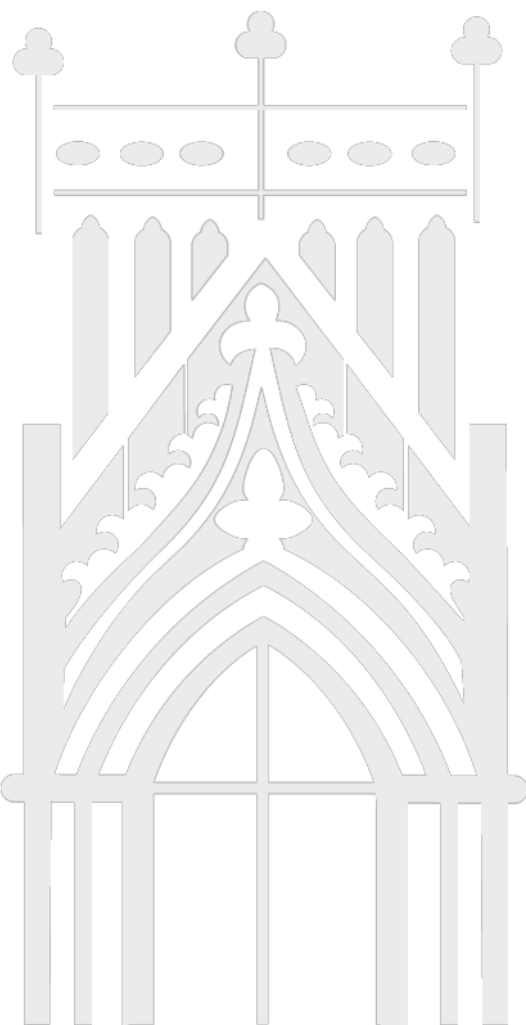


Mestrado em Computação Móvel

Desenvolvimento de Aplicações e Jogos para Alunos com Necessidades Especiais de Educação, utilizando como plataforma os Dispositivos Móveis.

Mateus Alves Victorelli

abril | 2019



Escola Superior
de Tecnologia e Gestão



INSTITUTO POLITÉCNICO DA GUARDA
Escola Superior de Tecnologia e Gestão

**Desenvolvimento de Aplicações e Jogos para Alunos com
Necessidades Especiais de Educação, utilizando como
plataforma os Dispositivos Móveis**

Projeto Aplicado do Mestrado em Computação Móvel

Mateus Alves Victorelli
Abril de 2019



INSTITUTO POLITÉCNICO DA GUARDA
Escola Superior de Tecnologia e Gestão

**Desenvolvimento de Aplicações e Jogos para Alunos com
Necessidades Especiais de Educação, utilizando como
plataforma os Dispositivos Móveis**

Relatório de projeto aplicado, submetido como requisito parcial
para obtenção do Grau de Mestre em Computação Móvel

Orientador: Professor Doutor Noel Lopes

Mateus Alves Victorelli
Abril de 2019

Agradecimentos

Aos professores presentes no curso, um especial obrigado ao Prof. Doutor Carlos Carreto e à Prof.^a Doutora Clara Silveira, pelo apoio durante o meu percurso académico.

Ao Prof. Doutor Noel Lopes pela orientação deste projeto.

À escritora Susana Campos, por autorizar a utilização da obra literária.

Ao ilustrador Rui Lobo, por autorizar a utilização de materiais gráficos.

Ao Centro de Recursos para a Inclusão Digital da Escola Superior de Educação e Ciências Sociais do Instituto Politécnico de Leiria (ESECS/IPL).

À Prof.^a Doutora Maria Fernanda Serrão Bastos de Oliveira, pelas orientações em temas relacionados com a educação especial.

Por fim um agradecimento especial às instituições e pessoas que contribuíram de alguma forma para a elaboração e divulgação deste projeto.

Resumo

Este relatório de projeto aplicado descreve como foi o processo de desenvolvimento de uma aplicação para dispositivos móveis (smartphones e tablets), na plataforma Android, destinada para os alunos na educação especial e para as pessoas com alguma deficiência, principalmente aquelas com deficiência intelectual, auditiva e surdez.

Observou-se uma oportunidade para a investigação e desenvolvimento de aplicações e jogos, utilizados para a diversão, entretenimento e educação. Direcionados para as crianças com necessidades especiais de educação e para as pessoas com deficiência, perturbações ou doenças crónicas.

Uma análise foi realizada para demonstrar o benefício da tecnologia na educação especial, além de mostrar alguns exemplos de produtos direcionados para este público.

Foi realizado um estudo para compreender as principais características e necessidades dos alunos na educação especial e das pessoas com deficiências, principalmente as pessoas com deficiência intelectual, auditiva e surdez, no qual este projeto evidenciou no desenvolvimento da aplicação.

Apresenta-se o processo de desenvolvimento de uma aplicação, mostrando os conceitos da engenharia de software como a arquitetura, requisitos, design, implementação, testes e resultados.

A aplicação desenvolvida neste projeto chama-se “O menino que tinha medo do escuro”, possui uma história infantil inclusiva com diferentes recursos para a comunicação, como a língua gestual portuguesa, pictogramas, textos, ilustrações, áudio com a locução e jogos utilizados para trabalhar a área cognitiva (memória, psicomotricidade, linguagem, imaginação, criatividade e desenho).

A aplicação “O menino que tinha medo do escuro” foi publicada nas principais lojas digitais, tendo até à data obtido mais de 9.000 instalações com avaliações e comentários positivos.

Mostra-se os planos para o trabalho futuro em relação à aplicação “O menino que tinha medo do escuro”, além de apresentar outras ideias para investigação e desenvolvimento de novas soluções nesta área.

Palavras-chave

Necessidades Especiais de Educação; Deficiências; Deficiência Intelectual; Deficiência Auditiva; Surdez; Dispositivos Móveis; Jogos; Aplicações; Inclusão; Android.

Abstract

This thesis describes a application development process, using mobile devices (smartphones and tablets) from the Android platform, intended for people with disabilities and children with special educational needs.

There was an opportunity to research and developed a new digital products (games and applications) focused on fun, entertainment and education, designed for people with disabilities, disorders or chronic conditions.

An analysis of the benefit to use technology in special education is presented, showing examples of products for this target audience.

A study is conducted about students in special education and people with disabilities, mainly with intellectual disability, hearing impairment and deafness.

The process of developing a software for the Android platform is presented, with the concepts of software engineering like architecture, requirements, design, implementation, testing and results.

The application “O menino que tinha medo do escuro” was developed, has a children’s story with different communication resources, like as the Portuguese sign language, pictograms, texts, illustrations, audio with the locution and games used to work in the cognitive areas (memory, psychomotricity, language, imagination, creativity and drawing).

The “O menino que tinha medo do escuro” application was published in the most important apps stores, having until the date more of 9.000 installations with positive ratings and comments.

The plans about future work of the “O menino que tinha medo do escuro” application is showed and new ideas to research and developed for this target audience.

Keywords

Special Needs Education; Disabilities; Intellectual Disability; Hearing Disability; Deafness; Mobile Devices; Games; Applications; Inclusion; Android.

Índice

1	Introdução	1
1.1	Propósito	1
1.2	Motivação	2
1.3	Desafios Identificados	3
1.4	Objetivos	4
1.5	Solução Proposta	5
1.6	Estrutura do Relatório de Projeto Aplicado	6
2	Estado da Arte	7
2.1	Necessidades Especiais de Educação	7
2.2	Tecnologias de Informação e Comunicação na Educação Especial	9
2.3	Dispositivos Móveis	13
2.4	Produtos Existentes para Educação Especial e Acessibilidade	20
3	Caracterização de Deficiência	47
3.1	Deficiência	47
3.2	Deficiência Intelectual	51
3.3	Deficiência Auditiva e Surdez	53
4	Desenvolvimento da Aplicação “O menino que tinha medo do escuro”	55
4.1	Solução Desenvolvida	55
4.2	Processo de Desenvolvimento de Software	57
4.2.1	Requisitos	58
4.2.2	Game Design	62
4.2.3	Arquitetura de Software	64
4.2.4	Plataforma Android	69
4.3	Implementação da Aplicação	70
4.3.1	Aplicação – O menino que tinha medo do escuro	84

4.3.2	Módulo – Livro Multimédia Básico	88
4.3.3	Módulo – Livro Multimédia Pictogramas	94
4.3.4	Módulo – Livro Multimédia Língua Gestual Portuguesa	100
4.3.5	Módulo – Jogo Desenhar	103
4.3.6	Módulo – Jogo da Memória	109
4.3.7	Módulo – Jogo do Puzzle	115
4.3.8	Módulo – Jogo Completar as Palavras	120
4.3.9	Módulo – Framework	128
4.3.10	Módulo – Recursos	143
5	Testes e Resultados	147
5.1	Testes	147
5.2	Publicações e Resultados	151
5.3	Empreendedorismo	154
5.4	Divulgações	157
5.5	Outros Projetos	163
6	Conclusão e Trabalho Futuro	167
6.1	Conclusão	167
6.2	Trabalho Futuro	168
	Bibliografia	171

Lista de Figuras

2.1	Exemplo dos símbolos pictográficos de comunicação	10
2.2	Tech Scan 8 digitalizador da fala	10
2.3	IntelliKeys USB	11
2.4	TrackerPro	11
2.5	Grid 3	12
2.6	MagicKey	12
2.7	IBM Simon	13
2.8	BlackBerry 5810	14
2.9	Apple iPhone	14
2.10	HTC Dream	15
2.11	Apple iPhone X, Google Pixel 2 XL e Samsung Galaxy S9+	15
2.12	PT Jaws	21
2.13	PT Focus 14	21
2.14	PT ZoomText	22
2.15	PT Smart Beetle	22
2.16	PT Locução Portátil	23
2.17	PT SPC	23
2.18	PT Grid 3	24
2.19	PT MagicContact	24
2.20	PT MagicKey	25
2.21	PT Magic Keyboard	25
2.22	PT Magic Eye	26
2.23	PT PC EYE Mini	26
2.24	Vox4all	27
2.25	Oscar Feels	27
2.26	Symbol LAB	28
2.27	Look to Learn	28

2.28	Mind Express 4	29
2.29	Comunicar com Símbolos	29
2.30	Boardmaker & Speaking Dynamically Pro	30
2.31	InVento 2	30
2.32	Jogos Memória – Junior	31
2.33	Letras e Palavras	31
2.34	Primeiros Anos Mix	32
2.35	MIND7	32
2.36	Proloquo2Go	33
2.37	TouchChat HD	33
2.38	Avaz Pro – AAC app for Autism	34
2.39	Social Skills With Billy	34
2.40	Autism/DTT Shapes	35
2.41	iCommunicate	35
2.42	iPrompts PRO	36
2.43	MyTalkTools	36
2.44	Abilipad	37
2.45	Speech FlipBook Standard	37
2.46	Voice4u AAC Communication	38
2.47	LetMeTalk	38
2.48	TalkTablet	39
2.49	Livox	39
2.50	Samsung LOOK AT ME	40
2.51	Montessori Words & Phonics	40
2.52	Autism Language and Cognitive Therapy with MITA	41
2.53	EKUI Alfabeto	41
2.54	Puzzles educativos para miúdo	42
2.55	Smile and Learn: Jogos educativos para crianças	42
2.56	O menino dos dedos tristes	43

2.57	Piu Caganita	44
2.58	A Rainha das Rosas	45
3.1	População portuguesa com dificuldade nas atividades diárias	49
4.1	Estrutura do documento Game Design Canvas	63
4.2	Documento Game Design Canvas da aplicação	64
4.3	Diferença entre os modelos monolítico e micro-serviços	65
4.4	Diferença entre as duas arquiteturas: monolítica e de micro serviços	66
4.5	Arquitetura monolítica da Uber	67
4.6	Arquitetura micro-serviços da Uber	68
4.7	Arquitetura desenvolvida para a aplicação	70
4.8	Criar projeto no Android Studio, passo inicial	72
4.9	Criar projeto no Android Studio, selecionar o tipo de projeto	73
4.10	Criar projeto no Android Studio, configuração inicial do projeto	73
4.11	Criar projeto no Android Studio, estrutura do projeto	74
4.12	Opção para executar a task install no gradle	76
4.13	Criar projeto híbrido no Android Studio, passo inicial criando a library . .	78
4.14	Criar módulo dentro do projeto no Android Studio	78
4.15	Estrutura do projeto no Android Studio com a library e a aplicação	80
4.16	Opções do gradle para compilar e executar projeto	82
4.17	Ecrã inicial e menu principal I	85
4.18	Ecrã inicial e menu principal II	85
4.19	Livro multimédia I	88
4.20	Livro multimédia II	89
4.21	Livro multimédia com os pictogramas I	95
4.22	Livro multimédia com os pictogramas II	95
4.23	Livro multimédia com a língua gestual portuguesa I	100
4.24	Livro multimédia com a língua gestual portuguesa II	100
4.25	Jogo desenhar I	103
4.26	Jogo desenhar II	104

4.27	Jogo desenhar III	104
4.28	Jogo da memória I	109
4.29	Jogo da memória II	109
4.30	Jogo da memória III	110
4.31	Jogo de quebra-cabeça I	115
4.32	Jogo de quebra-cabeça II	115
4.33	Jogo de quebra-cabeça III	116
4.34	Jogo de completar as palavras I	120
4.35	Jogo de completar as palavras II	121
4.36	Jogo de completar as palavras III	121
5.1	Apresentações com os utilizadores I	149
5.2	Apresentações com os utilizadores II	150
5.3	Reportagem no Programa Portugal em Direto – RTP	158
5.4	Apresentação no Agrupamento de Escolas Quinta de Marrocos	158
5.5	Reportagem no jornal O Interior	159
5.6	Reportagem no jornal Terras da Beira	160
5.7	Reportagem no jornal Diário de Viseu	161
5.8	Aplicação – Ernesto, Jogos para a Gaguez	163
5.9	Aplicação – Os Guerreiros do Pequeno Artur	164
5.10	Aplicação – Prancha Inclusiva Estações do Ano	165

Lista de Tabelas

2.1	Especificações dos Smartphones	16
2.2	Faturamento do mercado tecnológico em Portugal	18
2.3	Venda Mundial de dispositivos móveis por plataforma entre 2012 e 2017	19
2.4	Soluções oferecidas pela Fundação PT	20
3.1	População portuguesa com dificuldade nas atividades diárias	49
3.2	Alunos com necessidades especiais de educação ano letivo 2015/2016	50
5.1	Principais casos de testes	148
5.2	Instalações por país	152
5.3	Instalações por versão do sistema operativo Android	152
5.4	Instalações por modelo do dispositivo	152
5.5	Principais comentários das avaliações realizadas pelos utilizadores	153

Capítulo 1

Introdução

Este capítulo aborda os detalhes do projeto, explicando como surgiu a ideia, definindo o público-alvo e a plataforma utilizada. Apresenta as principais motivações para se desenvolver uma aplicação destinada às pessoas com deficiência e principalmente os alunos na educação especial. Demonstra os desafios identificados durante a investigação e os principais objetivos a serem alcançados, explicando como foi definida a solução. Mostra como está estruturado este relatório de projeto aplicado.

1.1 Propósito

O propósito inicial deste projeto foi o de utilizar as aplicações e jogos para os dispositivos móveis para reduzir uma desigualdade social.

Durante a fase inicial das pesquisas, procurou-se desenvolver um produto digital, sendo direcionado para as pessoas que por vários motivos sofre com a escassez de recursos lúdicos e didáticos. Neste contexto, o principal público-alvo são as crianças com necessidades especiais de educação e as pessoas com deficiências ou doenças crônicas.

A proposta deste projeto é desenvolver produtos (jogos e aplicações) para ensinar, divertir e entreter crianças com alguma deficiência ou doença crônica, sendo direcionados para a educação especial e com o objetivo de reduzir a carência de produtos e conteúdos para esta área.

A solução desenvolvida neste projeto será uma aplicação móvel para os dispositivos da plataforma Android, composta por um livro multimídia inclusivo e um conjunto de jogos lúdico-pedagógicos.

Pretende-se com esta aplicação enfatizar simultaneamente o lado lúdico com o didático e ser inclusivo, através de conteúdos acessíveis para os alunos na educação especial e também no ensino regular, transformando-se num instrumento essencial para desenvolver atividades de leitura, entretenimento e diversão com as crianças na idade pré-escolar e das que têm currículos adaptados através do ensino especial.

1.2 Motivação

A motivação para o desenvolvimento deste projeto deve-se à grande desigualdade que sofrem as pessoas com deficiência e principalmente os alunos na educação especial.

Nos últimos anos têm surgido várias medidas para alterar este cenário, principalmente através da preparação da sociedade para lidar com as pessoas com deficiência, criando novas soluções para ultrapassar as barreiras nesta área, mas é necessário um maior esforço para melhorar esta situação que permanece grave [Organização Mundial da Saúde, 2012].

As pessoas com deficiências tendem a apresentar uma pior perspectiva em relação à saúde, além de outros fatores como baixos níveis de escolaridade, carência económica e alta taxa de pobreza. A maior parte das pessoas com deficiência não possuem acesso adequado à saúde, à educação, tem menos oportunidades de emprego e sofrem de alguma forma um tipo de exclusão social [Organização Mundial da Saúde, 2012].

Atualmente existe pouca oferta de produtos e conteúdos direcionados para as pessoas com deficiências e para as crianças com necessidades especiais de educação. De acordo com o “Relatório Mundial Sobre a Deficiência” [Organização Mundial da Saúde, 2012]:

“As evidências empíricas disponíveis sugerem que pessoas com deficiências têm taxas de utilização das TIC significativamente menores que as pessoas sem deficiência. Em alguns casos, elas podem ser incapazes de ter acesso aos produtos e serviços mais básicos tais como telefones, televisão e Internet.”

A grande parte dos produtos disponíveis no mercado geralmente não estão acessíveis para este público, principalmente pela falta de disponibilidade em língua portuguesa, preços elevados praticados pelas empresas neste segmento, além da necessidade de equipamentos informáticos ou eletrónicos com um custo razoavelmente alto. É possível obter mais informações sobre este tema na secção “2.3 Soluções Existentes”.

Este cenário é motivado por vários fatores, como os problemas sociais, culturais e pelas próprias empresas e profissionais de diversas áreas que possuem uma grande dificuldade em compreender as necessidades e limitações destas pessoas e não conseguem criar meios para elaborar produtos mais acessíveis para este público.

Este projeto pretende desenvolver produtos digitais acessíveis (equipamentos de baixo custo) e inclusivos (com vários recursos de comunicação), utilizando como plataforma os dispositivos móveis (telemóveis, smartphones e tablets) que já estão presentes na vida de milhares de pessoas.

1.3 Desafios Identificados

Durante as investigações foram identificadas várias barreiras que este projeto pretende combater ou pelo menos amenizar. Consciencializados sobre os cenários que envolvem a inclusão e a educação especial, alguns dos desafios identificados foram:

- Afastamento das pessoas com deficiências ou de alunos na educação especial em relação ao uso da tecnologia no ensino.
- Pouco investimento em projetos, empresas e produtos para este público, a grande parte das empresas optam por não desenvolver produtos inclusivos ou direccionados para as pessoas com deficiências e para a educação especial por ser um mercado reduzido.
- Os produtos e equipamentos vocacionados para as pessoas com deficiências e para a educação especial geralmente possuem preços elevados, ver mais informações na secção “2.4 Produtos Existentes”.
- Muitas instituições que cuidam de pessoas com deficiências possuem dificuldades financeiras, não conseguindo suportar o custo dos produtos existentes no mercado sem a ajuda de doações.
- Os produtos desenvolvidos por empresas internacionais geralmente não têm suporte para a língua portuguesa.

O “Projeto-Lei n.º 329/XIII/2.^a” notificou que existem cerca de 1 milhão de pessoas com deficiências em Portugal, teoricamente não existe nenhuma inibição para estas pessoas frequentarem o ensino superior, mas a maior parte das instituições de ensino não fornecem as condições necessárias para isto acontecer. De acordo com um inquérito realizado em 291 instituições do ensino superior constatou-se que apenas 94 têm serviços de apoio para pessoas com deficiências [Silva, 2017].

Durante o ano letivo (2016/2017) foram disponibilizadas 1.000 vagas destinadas às pessoas com deficiências no ensino superior, sendo que apenas 14% destas vagas foram preenchidas. Dos 11.062 alunos com necessidades especiais de educação que frequentaram as escolas regulares de ensino, apenas 140 ingressaram no ensino superior durante o ano letivo (2017/2018) [GEP, 2016].

Em relação ao mercado de trabalho das pessoas com deficiências, a taxa de desemprego é extremamente alta comparada com a média da população, chegando a ser 3 vezes superior comparativamente com aquelas que não possuem nenhuma incapacidade [Ferreira, 2017].

1.4 Objetivos

Este projeto pretende desenvolver uma aplicação para os dispositivos móveis (telemóvel, smartphone e tablet), na plataforma Android e com as seguintes características:

- Ser uma história infantil inclusiva, utilizando recursos de comunicação como língua gestual portuguesa, símbolos pictográficos de comunicação, textos, ilustrações e áudio com locução.
- Conter jogos lúdico-pedagógicos utilizados para desenvolver as capacidades cognitivas (memória, psicomotricidade, linguagem, imaginação, criatividade e desenho).
- Ser direcionada para o público infantil.

A escolha deste tema é motivada pelo facto de até o início das investigações deste projeto (ano 2012) não existir nenhuma solução semelhante no mercado de aplicações móveis. Pretende-se comprovar que é possível desenvolver uma aplicação inclusiva com um esforço reduzido, contando com o apoio de pessoas especializadas em determinadas áreas e com um processo de engenharia de software simplificado.

O objetivo principal é desenvolver um produto digital acessível e inclusivo para os alunos na educação especial e pessoas com deficiências, utilizando os dispositivos móveis como plataforma. A seguir estão os objetivos a serem alcançados neste projeto:

- Melhorar a qualidade de vida das pessoas com deficiência, por meio da introdução de novas soluções digitais direcionadas para o ensino, entretenimento e diversão.
- Proporcionar mais oportunidades para o autodesenvolvimento, enriquecimento das capacidades cognitivas, comunicativas, aptidões sensoriais, comportamento social e emocional.
- Disponibilizar ferramentas para auxiliar os profissionais e familiares de pessoas com deficiências.
- Reduzir o distanciamento dos produtos tecnológicos em relação às pessoas com deficiências e aos alunos na educação especial.
- Tornar a tecnologia mais acessível, informando e incentivando sobre a importância de caminharmos para a inclusão.

1.5 Solução Proposta

Durante as investigações foi identificada a existência de diversas aplicações para a educação especial e pessoas com deficiência, porém o preço praticado pelas empresas deste segmento é geralmente elevado, fazendo com que estas soluções não estejam ao alcance deste público.

Identificou-se que existem muitos produtos direcionados para as pessoas com problemas de comunicação, principalmente para a comunicação alternativa e aumentativa. Outra categoria de produtos que chamaram a atenção foram os livros inclusivos ou em multiformatos.

Num primeiro momento surgiu a ideia de desenvolver um software para a comunicação alternativa e aumentativa, devido aos preços elevados e alguns produtos necessitarem de equipamentos com custo elevado (computadores, dispositivos eletrônicos, etc.) comparados com os smartphones e tablets.

Porém, chegou-se à conclusão que é necessário um grande esforço para criar um software para auxiliar a comunicação de pessoas com deficiências e alunos na educação especial, motivado principalmente pela necessidade da ajuda dos profissionais de áreas como a educação especial, reabilitação e terapia da fala.

Em relação aos livros inclusivos, as soluções existentes estão disponíveis em formatos convencionais, como os livros impressos e em braile, textos e ilustrações adaptadas para baixa visão, pictogramas, locução e vídeo com a tradução para língua gestual.

Comprovou-se a inexistência de aplicações móveis com o formato de um livro multimédia inclusivo. Neste contexto, verificou-se que com a exceção do braile é possível desenvolver uma aplicação móvel com o suporte de diferentes formatos para a comunicação.

A proposta deste projeto foi desenvolver uma aplicação móvel chamada “O menino que tinha medo do escuro”, utilizada para comprovar que é possível criar uma solução inclusiva e acessível, que utilize os recursos disponíveis nos dispositivos móveis. Será direcionada principalmente para o público infantil e utiliza diferentes recursos de comunicação como textos e ilustrações adaptados, Símbolos Pictográficos de Comunicação (SPC), áudio com a locução da história e o vídeo com a tradução para língua gestual portuguesa.

A plataforma móvel selecionada para suportar este projeto é a Google Android, motivado pela grande quantidade de dispositivos existentes no mercado, com enorme diversidade de equipamentos e preços acessíveis.

1.6 Estrutura do Relatório de Projeto Aplicado

Este relatório de projeto aplicado está estruturado em seis capítulos, sendo a introdução, estado da arte, caracterização de deficiência, desenvolvimento da aplicação, resultados e por fim a conclusão e o trabalho futuro.

O capítulo 1 (Introdução) contém as informações que justificam a escolha deste tema de projeto. Pretende-se demonstrar quais são as barreiras identificadas e quais serão os objetivos a serem alcançados, além de definir como será a solução desenvolvida neste projeto.

O capítulo 2 (Estado da Arte) contém uma análise para verificar o benefício da tecnologia na educação especial. Demonstrem-se estimativas sobre como os dispositivos móveis estão inseridos no quotidiano das pessoas. Serão apresentados exemplos de produtos direcionados para os alunos na educação especial e pessoas com deficiências.

O capítulo 3 (Caracterização de Deficiência) apresenta-se uma descrição sobre este tema. Abordar-se-á a deficiência intelectual e auditiva, que são as principais deficiências que este projeto evidencia. Serão demonstradas estimativas sobre o total de pessoas com deficiências e os alunos matriculados na educação especial.

O capítulo 4 (Desenvolvimento da Aplicação “O menino que tinha medo do escuro”) descreve-se como foi o processo da análise e o desenvolvimento da solução proposta, além de conceitos sobre a engenharia de software, definição dos requisitos, design, arquitetura de software, implementação, testes e os resultados obtidos.

O capítulo 5 (Resultados) apresentam-se os resultados obtidos da aplicação publicada nas lojas digitais. Abordam-se outros projetos desenvolvidos no contexto deste tema. Explora a relação deste projeto com a vertente do empreendedorismo e demonstram-se as divulgações que ocorreram nos meios de comunicação.

O capítulo 6 (Conclusão e Trabalho Futuro) compreende os resultados obtidos durante o desenvolvimento deste projeto aplicado e sobre o trabalho futuro.

Capítulo 2

Estado da Arte

Neste capítulo são apresentadas as necessidades especiais de educação (NEE), explicando as principais dificuldades dos alunos na educação especial. Demonstra a importância das Tecnologias da Informação e Comunicação (TIC) na educação especial, além de apresentar exemplos de soluções utilizadas nesta área. Mostra alguns produtos para educação especial com o objetivo de verificar o que existe no mercado e suas principais características. Aborda a evolução dos dispositivos móveis, principalmente os smartphones que atualmente se tornaram um equipamento popular.

2.1 Necessidades Especiais de Educação

Os alunos com necessidades especiais de educação são aqueles que apresentam uma incapacidade que reduz o desempenho nas áreas relacionadas com a aprendizagem. São resultantes de deficiências ou limitações, de ordem sensorial, motora, intelectual, além de distúrbios da fala e linguagem, perturbações graves da personalidade, comportamento e problemas de saúde. Um aluno com necessidades especiais de educação irá apresentar problemas relacionados com a aprendizagem durante toda a fase de escolarização, necessitando de uma maior atenção por parte do corpo docente, além de recursos educacionais suplementares [Oliveira, 2013].

O total de alunos na educação especial em Portugal matriculados durante o ano letivo 2015/2016 foram 78.175 alunos, sendo 3.573 no ensino pré-escolar, 63.540 no básico e 11.062 no secundário [GEP, 2016].

Através do Dec. Lei N.º 319/91, 23 de agosto, a política educativa tornou-se integrada no ensino de alunos com necessidades especiais de educação, passando estes terem o direito de frequentar uma turma de ensino regular. Foram introduzidos vários princípios para criar uma educação mais integrada e inclusiva, como [Rodrigues e Nogueira, 2010]:

- Obtenção de um amplo conhecimento sobre a situação escolar e sociofamiliar de cada aluno com necessidades especiais de educação.
- Realização de uma planificação educacional individualizada, flexível e adaptada para cada situação.
- Inserção dos professores de educação especial como um recurso da escola.
- Alteração da nomenclatura “alunos com deficiência” para “alunos com necessidades especiais de educação”, introduzindo o conceito por completo e não apenas uma mudança de designação.
- Suporte legal para a aplicação de práticas pedagógicas diferenciadas que entram em rutura comparada com os padrões tradicionais de ensino.
- Regulamentação de medidas que visam proporcionar melhores condições de ensino, possibilitando uma flexibilização de currículos e avaliações, através dos Programas Educativos Individuais – PEI.

A legislação atual tem como base o Dec. Lei N.º 3/2008, que revogou a anterior que estava em vigor há 19 anos (Dec. Lei N.º 319/91). A principal mudança foi permitir aos indivíduos abrangidos pelos serviços de educação especial a Classificação Internacional de Funcionalidade – CIF, estabelecida pela Organização Mundial de Saúde. Alguns dos princípios regulamentados neste decreto de lei foram [Rodrigues e Nogueira, 2010]:

- Expansão no âmbito da educação especial ao setor privado, cooperativo e pré-escolar, além do ensino básico e secundário que já estavam abrangidos pelo decreto anterior.
- Ênfase na necessidade de uma regulamentação dos instrumentos de estudos, com medidas mais específicas aplicadas ao aluno.
- Uma maior participação dos encarregados de educação, reforçando o poder de decisão na coordenação, avaliação e planificação.

2.2 Tecnologias de Informação e Comunicação na Educação Especial

A utilização das TIC (Tecnologias da Informação e Comunicação) possui um grande potencial na educação especial, facilitando os esforços necessários para inclusão dos alunos com necessidades especiais de educação nas salas de aulas. As ferramentas e os conteúdos são essenciais para promover uma melhor educação e facilitar a acessibilidade, inclusão e integração de alunos e pessoas com deficiências [Istemic Starcic e Bagon, 2014].

Atualmente as TIC estão inseridas no quotidiano de muitas pessoas, fornecendo ferramentas para promover uma melhor qualidade de vida, como reduzir a exclusão social das pessoas com deficiência, além de serem um excelente instrumento utilizado na educação especial.

Um dos principais objetivos das TIC na educação especial é proporcionar a igualdade educacional, fornecendo meios para auxiliar o processo de aprendizagem dos alunos com necessidades especiais de educação [Watkins, 2013].

Nos últimos anos surgiram soluções direcionadas para o apoio e reabilitação das pessoas com deficiências ou limitações, principalmente nas áreas da mobilidade, comunicação, manipulação e orientação. Estas soluções são chamadas de comunicação alternativa e aumentativa [Azevedo, 2005].

Uma das principais soluções existentes para os sistemas de comunicação alternativos e aumentativos é o SPC – Símbolo Pictográfico de Comunicação. Este sistema foi desenvolvido em 1981 pela terapeuta da fala norte-americana Roxanna Mayer Johnson, sendo introduzido em Portugal nos anos 90, através da tradução dos pictogramas para a língua portuguesa [Campina, 2016].

O Símbolo Pictográfico de Comunicação é uma ferramenta utilizada para facilitar o processo de comunicação e aprendizagem das pessoas com dificuldade e/ou ausência da fala ou sem escrita funcional. Este sistema é usado por pessoas com deficiências e limitações associadas a um défice cognitivo, como a deficiência intelectual, paralisia cerebral, afasia, apraxia e disartria, acidentes vasculares e limitações resultantes de um pós-operatório. Atualmente é considerado como o sistema de comunicação aumentativa e alternativa mais utilizado no mundo, devido à sua simplicidade de uso, pouca exigência cognitiva e fácil adequação para qualquer faixa etária [Campina, 2016].

A figura 2.1 foi utilizada na história “O menino que tinha medo do escuro”, demonstra um exemplo dos símbolos pictográficos de comunicação:



Figura 2.1: Frase a utilizar o formato dos símbolos pictográficos de comunicação

Os principais benefícios da utilização dos símbolos pictográficos para a comunicação são [Campina, 2016]:

- Grande facilidade no processo de aprendizagem.
- Adequado para ser utilizado por pessoas de qualquer idade.
- Elevada diferenciação entre os símbolos gráficos para facilitar a compreensão e evitar que as pessoas confundam o significado dos pictogramas.
- Contém um conjunto das principais palavras e ações mais utilizadas no dia a dia.
- É facilmente reproduzido no formato digital ou com impressões em papel.

Existem diversos dispositivos e software para a comunicação alternativa e aumentativa, a seguir serão citados alguns exemplos [Azevedo, 2005]:

- **Digitalizadores de Fala:** São os equipamentos que fornecem funcionalidades para gravar e ouvir a voz ou outros sons, permitindo que o utilizador sem a comunicação oral selecione na interface (por toque, movimento ou até mesmo por sopro) um determinado elemento. A figura 2.2 demonstra este equipamento.



Figura 2.2: Tech Scan 8 digitalizador da fala ¹

¹ **Tech Scan 8:** https://www.amdi.net/index.php?_route_=tech-scan-8, acedido em 20/11/2018.

- **Teclado de conceitos:** São teclados inteligentes, programáveis e sensíveis ao toque ou com auxílio de apontadores. Fornecem uma maior facilidade na interação com o computador. Contêm um conjunto de opções chamadas grelhas, sendo pré-concebidas para facilitar a introdução de informação pelo utilizador. A figura 2.3 demonstra este equipamento.



Figura 2.3: IntelliKeys USB ²

- **Dispositivo apontador:** É uma interface de interação com o computador, fornece um apontador para as pessoas com os movimentos limitados. Utiliza um emissor e um recetor de sinais, controlado por refletores de movimentos, por exemplo, um sensor que usa a movimentação da cabeça do utilizador para movimentar um rato no computador. A figura 2.4 demonstra este equipamento.



Figura 2.4: TrackerPro ³

² **IntelliKeys USB:** <https://www.bltt.org/hardware/intellikeys.htm>, acedido em 20/11/2018.

³ **TrackerPro:** <https://www.ablenetinc.com/trackerpro>, acedido em 20/11/2018.

- **Software GRID:** É um sistema de teclados por toque no ecrã adequado para as pessoas com alguma limitação cognitiva ou motora. O seu funcionamento pode ser considerado como um simulador do teclado convencional, substituindo quase por completo as funcionalidades existentes no computador. A figura 2.5 demonstra este sistema.



Figura 2.5: Grid 3 ⁴

- **MagicKey:** Este software fornece o controlo de um rato para o computador através do olhar, permitindo ao utilizador usar qualquer aplicação de um computador. Através dos movimentos dos olhos é possível efetuar todos os cliques existentes no rato, inclusive a opção de rolar uma página na vertical. A figura 2.6 demonstra este software.



Figura 2.6: MagicKey ⁵

⁴ **Grid 3:** <https://thinksmartbox.com/product/grid-3>, acedido em 20/11/2018.

⁵ **MagicKey:** <http://www.magickey.ipg.pt/magickey.html>, acedido em 20/11/2018.

2.3 Dispositivos Móveis

Os dispositivos móveis são utilizados quotidianamente por milhares de pessoas em todo o mundo, tornaram-se uma ferramenta tecnológica portátil e móvel com uma infinidade de funcionalidades para os utilizadores [Alqahtani e Mohammad, 2015].

A história dos smartphones não é antiga, o primeiro dispositivo denominado smartphone surgiu em 1993, sendo o modelo “Simon” que foi desenvolvido pela IBM. Podemos considerar como a primeira evolução dos smartphones quando a empresa RIM lançou em 2002 o modelo “BlackBerry 5810”, introduzindo várias funcionalidades inovadoras para esta época, como o acesso à internet, email, câmara fotográfica e vídeo, etc. Mas a grande revolução foi em 2007 com o lançamento do “iPhone” desenvolvido pela Apple e com o sistema operativo Google Android [Saha e Saha, 2018].

O sistema operativo Android proporcionou para os diversos fabricantes de smartphones a possibilidade de construir dispositivos diversificados (dimensões, ecrãs, processador, memória, câmara fotográfica, sensores, etc.), atendendo às necessidades e à preferência dos consumidores. Em 2008 surgiu o primeiro dispositivo da plataforma Android, o modelo “HTC Dream”. A partir de 2008 os smartphones se tornaram mais acessíveis, antigamente estes dispositivos eram utilizados no segmento empresarial devido ao custo elevado [Saha e Saha, 2018].

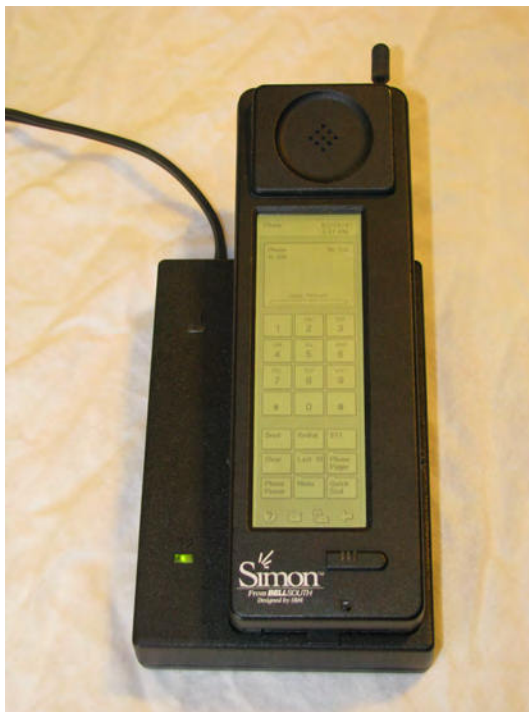


Figura 2.7: IBM Simon ⁶

⁶ **IBM Simon:** <https://habr.com/company/ibm/blog/184490/>, acedido em 20/11/2018



Figura 2.8: BlackBerry 5810 ⁷



Figura 2.9: Apple iPhone ⁸

⁷ **BlackBerry 5810:** <http://www.blackberry.com/news/connection/200201.shtml>, acessado em 20/11/2018

⁸ **iPhone:** <https://www.apple.com/br/newsroom/2007/01/09Apple-Reinvents-the-Phone-with-iPhone>, acessado em 20/11/2018



Figura 2.10: HTC Dream ⁹



Figura 2.11: Apple iPhone X, Google Pixel 2 XL e Samsung Galaxy S9+ ¹⁰

⁹ **HTC Dream:** <https://support.t-mobile.com/community/phones-tablets-devices/android/t-mobile-g1>, acedido em 20/11/2018

¹⁰ **GSMARENA:** <https://www.gsmarena.com/compare.php3?idPhone1=8858&idPhone2=8720&idPhone3=8967>, acedido em 20/11/2018

As figuras anteriores (2.7, 2.8, 2.9, 2.10 e 2.11) ilustram como eram os primeiros smartphones, além dos modelos atuais. Neste contexto, a tabela 2.1 demonstra as principais especificações dos dispositivos citados anteriormente, o objetivo é demonstrar a grande evolução tecnológica que existiu neste segmento.

Tabela 2.1: Especificações dos Smartphones [Anh, 2016]

Dispositivo	Processador	RAM	Armazenamento	Ecrã	SO
Simon	Intel NEC V20 16 MHz	1 MB	1 MB	160 x 293	ROM-DOS
BlackBerry 5810	ARM 7EJ-S 260 MHz	1 MB	8 MB	360 x 340	BlackBerry OS
iPhone	Samsung 76JZ 412 MHz	128 MB	4 GB	320 x 480	Apple iOS 1
HTC Dream	MSM7201A 528 MHz	192 MB	16 GB	320 x 480	Android 1.6
iPhone X	A11 Bionic 2.39 GHz	3 GB	64 / 256 GB	1125 x 2436	iOS 11.x
Pixel 2 XL	Snapdragon 835 2.35 Ghz	4 GB	64 / 128 GB	1440 x 2880	Android 8
Galaxy S9+	Snapdragon 845 2.7 GHz	6 GB	64 / 512 GB	1440 x 2960	Android 8

A era do smartphones começou realmente em 2007 quando a Apple anunciou o iPhone e ao mesmo tempo a Google lançou o sistema operativo Android que teve um grande crescimento com o suporte de vários fabricantes. Antigamente existia uma diferença entre telemóveis e smartphones, atualmente os dispositivos mais vendidos são os smartphones, com vários fabricantes competindo entre si no mercado, como Apple, Google, Samsung, LG, Lenovo, Asus, Xiaomi, etc. [Anh, 2016].

Atualmente diversos autores afirmam que os smartphones são tão poderosos que estão a substituir os computadores pessoais, principalmente pela ampla gama de aplicações que têm impacto em vários aspetos da nossa vida, desde o entretenimento, ensino, comunicação, saúde e negócios [Ahmad et al., 2018].

Com a introdução das ferramentas para o desenvolvimento de software e a criação das lojas de aplicações, milhares de empresas e desenvolvedores tiveram a oportunidade de distribuir as suas aplicações para os mais diferentes fins. Em 2015 o total de aplicações disponíveis nas lojas Google Play Store e na Apple Store somaram um total superior a 1 milhão de aplicações e em 2017 a receita gerada ultrapassou os 70 bilhões de dólares [Shen et al., 2017].

Os dispositivos móveis com o sistema operativo Google Android possuem vantagens para a utilização das TIC na educação, comparativamente com o iOS. As principais vantagens são [Trabelsi et al., 2017]:

- Constante crescimento e aprimoramento, proporcionando inovação no hardware e software, além da existência de uma grande variedade de dispositivos.
- São utilizados por milhares de pessoas, principalmente pelos estudantes.
- Possui um sistema operativo open-source, baseado no Linux.
- Fornece uma variedade de aplicações, ferramentas para o desenvolvimento e documentação.
- Os dispositivos Android podem-se tornar uma ferramenta pedagógica acessível num contexto educacional moderno.

O mercado dos dispositivos móveis está em constante crescimento, principalmente no segmento das telecomunicações. Segundo o relatório “The Mobile Economy 2016” as comunicações móveis ultrapassaram o total de 7.6 bilhões de serviços ativados, totalizando cerca de 4.7 bilhões de utilizadores únicos, este valor ultrapassou o total da população mundial [Association, 2016].

O estudo realizado pela empresa GfK TEMAX®¹¹ informou que o total de smartphones vendidos em 2015 atingiu a marca de 1.3 bilhões de dispositivos, proporcionando uma receita de 399.2 bilhões de dólares americanos [Rogers, 2016].

¹¹ **Descrição:** Índice desenvolvido pela GfK com o objetivo de monitorizar o mercado dos produtos tecnológicos.

Em Portugal, a Autoridade Nacional de Comunicações disponibilizou o relatório “Serviços Móveis – Informação Estatística, 3.º Trimestre de 2015”¹² com as informações deste segmento. A seguir são citados os principais indicadores deste relatório:

- O número de dispositivos habilitados a utilizar o serviço móvel atingiu a marca de 16,7 milhões.
- O serviço móvel ultrapassou a média de 161,4 por 100 habitantes, com uma taxa de utilização efetiva de 123,2 por 100 habitantes.
- No último trimestre de 2015, cerca de 95,1% do total de residentes em Portugal eram clientes do serviço telefónico móvel. Esta alta taxa é justificada por fatores como utilizadores com mais de um serviço móvel e pelos serviços móveis do segmento empresarial.
- O crescimento da utilização de smartphones, cerca de 5.5 milhões dos residentes em Portugal usam estes dispositivos.

Um estudo realizado sobre o terceiro trimestre de 2015, informou que o mercado dos produtos tecnológicos em Portugal apresentaram resultados negativos em quase todos os setores, com exceção das telecomunicações. A faturação deste segmento ultrapassou a marca de 500 milhões de Euros no 3.º trimestre de 2015, com grande destaque para os smartphones que representaram um valor superior a 35% do total das vendas neste segmento de mercado, conforme demonstra a tabela 2.2 [Figueiredo, 2015]:

Tabela 2.2: Faturamento do mercado tecnológico em Portugal (em milhões de Euros) [Figueiredo, 2015]

	Q1	Q2	Q3	Q1 – Q3
	2015	2015	2015	2015
Telecomunicações	165	159	191	515
Tecnologias de Informação	127	110	131	368
Grandes Eletrodomésticos	99	104	125	327
Eletrónica de Consumo	85	77	94	256
Pequenos Eletrodomésticos	43	39	47	130
Equipamento Escritório	26	25	23	74
Fotografia	11	13	14	38
TOTAL	557	527	626	1.709

¹² **Serviços Móveis 2015 – ANACOM:** https://www.anacom.pt/streaming/STMoveis3Trim2015.pdf?contentId=1373199&field=ATTACHED_FILE, acedido em 20/11/2018

Na última década o uso dos dispositivos móveis aumentou consideravelmente, devido à popularidade dos smartphones e ao crescimento da utilização das aplicações móveis. Este cenário tende a acentuar-se graças aos novos dispositivos e aplicações que surgem constantemente no mercado [Trabelsi et al., 2017].

As tecnologias móveis nos últimos anos tornaram-se muito populares, principalmente pela introdução dos smartphones no quotidiano da maioria das pessoas. Através das aplicações para os dispositivos móveis, surgiram uma infinidade de soluções para educação, diversão, entretenimento e no trabalho [Alqahtani e Mohammad, 2015].

As plataformas Google Android e Apple iOS possuem a maior quota do mercado de dispositivos móveis, em 2017 o total chegou a ser 99,8%, conforme demonstra a tabela 2.3 com as vendas de smartphones por plataforma nos últimos anos [Dutta et al., 2017]:

Tabela 2.3: Venda Mundial de dispositivos móveis por plataforma entre 2012 e 2017 [Dutta et al., 2017]

Ano	Android	iOS	Windows Phone	BlackBerry	Outros
2017	86,1%	13,7%	N/A	N/A	0,2%
2016	84,1%	14,8%	0,7%	0,2%	0,2%
2015	78,0%	17,5%	3,6%	0,4%	0,5%
2014	81,2%	15,2%	2,5%	0,4%	0,7%
2013	75,5%	16,9%	3,2%	2,9%	1,5%
2012	59,2%	22,9%	2,0%	6,4%	9,5%

2.4 Produtos Existentes para Educação Especial e Acessibilidade

Em Portugal a Fundação PT é uma das principais entidades responsáveis por fornecer produtos destinados à educação, saúde e acesso às comunicações. Esta fundação comercializa um conjunto de software desenvolvido por outras empresas com o objetivo de promover a inclusão social e melhorar a qualidade de vida das pessoas com necessidades especiais. A seguir na tabela 2.4 estão as soluções existentes de acordo com cada determinada área:

Tabela 2.4: Soluções oferecidas pela Fundação PT

Software	Visual	Fala / Comunicação	Cognitiva	Auditiva / Surdez	Neuromotora
PT Jaws	X				
PT Focus 14	X				
PT ZoomText	X				
PT Smart Beetle	X				
PT Locução Portátil	X				
PT SPC		X	X		X
PTGrid 3		X	X	X	X
PT MagicContact		X	X		X
PT Magic Keyboard			X		X
PT Magic Key			X		X
PT Magic Eye					X
PT PC EYE Mini					X

A seguir serão apresentadas informações dos produtos anteriormente citados na tabela 2.4:

PT Jaws: É um software com a finalidade de converter em voz o texto apresentado no ecrã de um computador, é destinado para as pessoas cegas, sendo compatível com o sistema operativo Windows e com suporte do idioma português. É desenvolvido pela empresa norte-americana Freedom Scientific e atualmente é considerado como um dos leitores mais utilizado pelas pessoas cegas. O preço é 2.100€ e para os clientes com necessidades especiais é 600€. A figura 2.12 demonstra este software.



Figura 2.12: PT Jaws ¹³

PT Focus 14: É um equipamento portátil que permite a leitura e escrita em braile nos smartphones e tablets, sendo compatível com o sistema operativo Android e iOS. É desenvolvido pela empresa norte-americana Freedom Scientific e o preço é 1.559€ e para os clientes com necessidades especiais é 450€. A figura 2.13 demonstra este equipamento.



Figura 2.13: PT Focus 14 ¹⁴

¹³ **PT Jaws:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTJaws.aspx>, acedido em 20/11/2018

¹⁴ **PT Focus 14:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTFocus14.aspx>, acedido em 20/11/2018

PT ZoomText: É um software desenvolvido pela empresa norte-americana VFO com a função em ampliar o ecrã do computador e converter o texto para a voz, é destinado para as pessoas com baixa-visão, é compatível com o sistema operativo Windows. O preço é 1.060€ e para os clientes com necessidades especiais é 300€. A figura 2.14 demonstra este software.

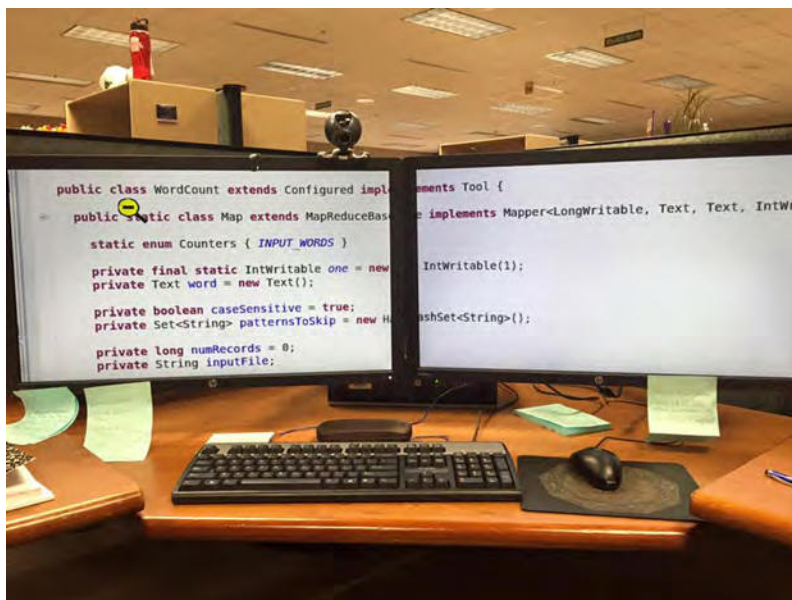


Figura 2.14: PT ZoomText ¹⁵

PT Smart Beetle: É um equipamento portátil para a leitura e escrita em braille nos smartphones, tablets e computadores, é compatível com os sistemas operativos Android, iOS, Windows e OS X, foi desenvolvido pela empresa norte-americana HIMS Inc. O preço é 1.240€ e para os clientes com necessidades especiais é 350€. A figura 2.15 demonstra este equipamento.



Figura 2.15: PT Smart Beetle ¹⁶

¹⁵ **PT ZoomText:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTZoomText.aspx>, acedido em 20/11/2018

¹⁶ **PT Smart Beetle:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTSmartBeetle.aspx>, acedido em 20/11/2018

PT Locução Portátil: É um software com um conversor de texto para voz que utiliza um sintetizador de voz portuguesa e é compatível com o NVDA (Non Visual Desktop Access que é um software gratuito de leitura do ecrã) e com o sistema operativo Windows. O preço é 95€ e para os clientes com necessidades especiais é 30€. A figura 2.16 demonstra este software.



Figura 2.16: PT Locução Portátil ¹⁷

PT SPC: É uma licença para utilizar um conjunto de 4.800 símbolos pictográficos de comunicação dentro dos software PT GRID 2 e PT GRID 3. O preço é 89€ e para os clientes com necessidades especiais é 50€. A figura 2.17 demonstra este software.

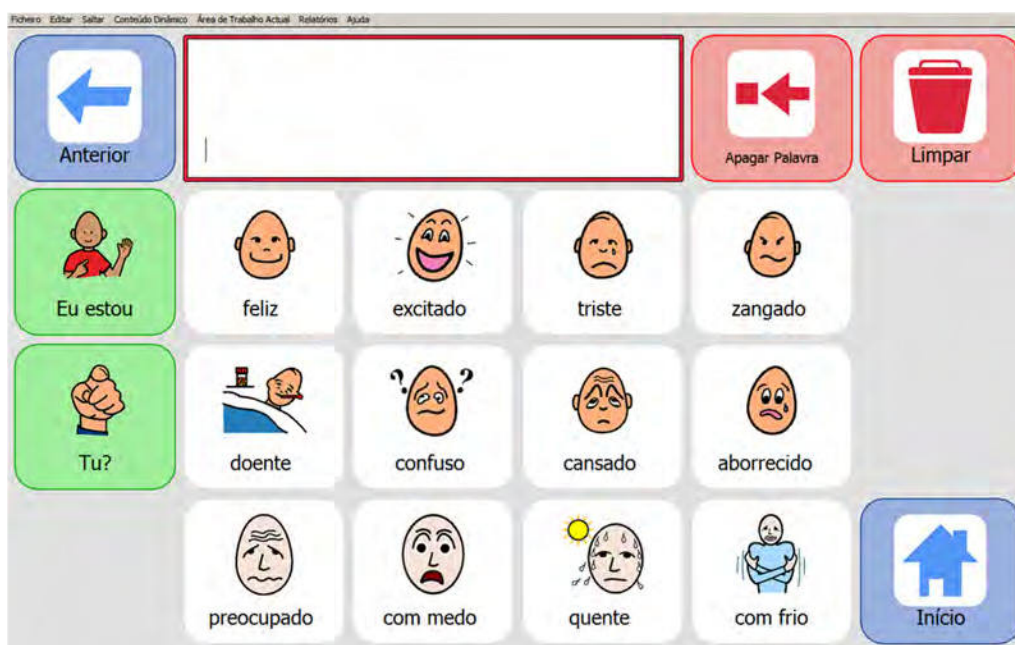


Figura 2.17: PT SPC ¹⁸

¹⁷ **PT Locução Portátil:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTLocuçãoPortátil.aspx>, acedido em 20/11/2018

¹⁸ **PT SPC:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTSPC.aspx>, acedido em 20/11/2018

PT Grid 3: É um software compatível com o sistema operativo Windows, utilizado para a comunicação alternativa e aumentativa, possui um sistema de teclados por toque no ecrã, adequado a pessoas com limitações cognitivas e motoras. É dividido em vários módulos que deverão ser adquiridos separadamente. O preço é 599€ e para os clientes com necessidades especiais é 180€. A figura 2.18 demonstra este software.



Figura 2.18: PT Grid 3 ¹⁹

PT MagicContact: É uma aplicação gratuita para sistema operativo Android, possui funcionalidades para promover a acessibilidade nos dispositivos móveis (smartphones e tablets). É destinada às pessoas com limitação motora e foi desenvolvida pela empresa MagicKey em parceria com a Fundação PT. A figura 2.19 demonstra esta aplicação móvel.



Figura 2.19: PT MagicContact ²⁰

¹⁹ **PT Grid 3:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTGrid3.aspx>, acedido em 20/11/2018

²⁰ **PT MagicContact:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTMagicContact.aspx>, acedido em 20/11/2018

PT MagicKey: É um sistema desenvolvido pela empresa MagicKey que controla o rato do computador através dos movimentos da cabeça. Permite ao utilizador usar qualquer aplicação de um computador e efetuar todos os cliques existentes no rato, inclusive a opção de rolar uma página na vertical. O preço é 1009€ e para os clientes com necessidades especiais é 359€. A figura 2.20 demonstra este sistema.

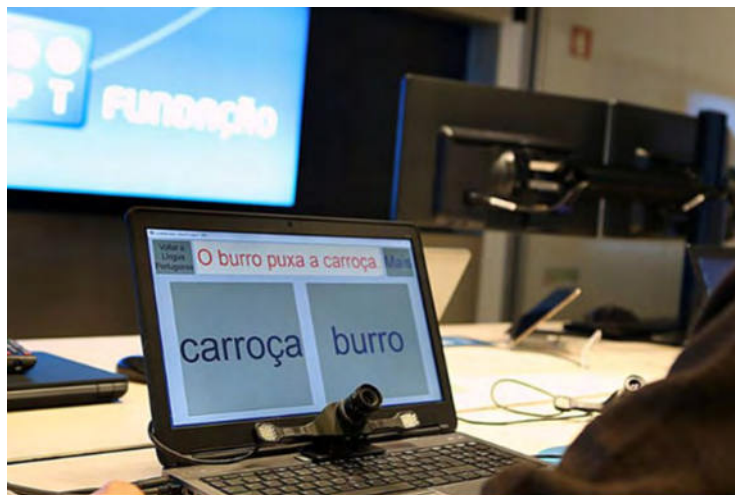


Figura 2.20: PT MagicKey ²¹

PT Magic Keyboard: É um sistema complementar para os produtos PT MagicKey e PT Magic Eye, desenvolvido pela empresa MagicKey, fornece uma forma para controlar o computador através de movimentos da cabeça ou dos olhos, apresenta teclados configuráveis e é destinado para as pessoas com limitações motoras. O preço é 159€ e para os clientes com necessidades especiais é 55€. A figura 2.21 demonstra este sistema.

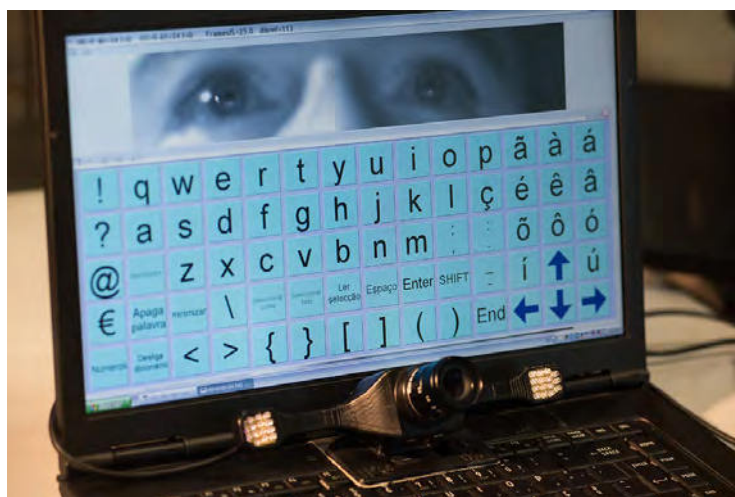


Figura 2.21: PT Magic Keyboard ²²

²¹ **PT MagicKey:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTMagicKey.aspx>, acedido em 20/11/2018

²² **PT Magic Keyboard:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTMagicKeyboard.aspx>, acedido em 20/11/2018

PT Magic Eye: É um sistema desenvolvido pela empresa MagicKey, permite controlar o cursor de um rato para computador através dos movimentos do olhar. É destinado para as pessoas com limitações motoras. O preço é 1.800€ e para os clientes com necessidades especiais é 599€. A figura 2.22 demonstra este sistema.



Figura 2.22: PT Magic Eye ²³

PT PC EYE Mini: É uma câmara desenvolvida pela empresa norte-americana Tobii Dynavox, permite aos utilizadores com limitações neuromotoras usar o computador através do olhar, sendo compatível com o sistema operativo Windows. O preço é 1.619€ e para os clientes com necessidades especiais é 560€. A figura 2.23 demonstra esta câmara.



Figura 2.23: PT PC EYE Mini ²⁴

²³ **PT Magic Eye:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTMagicEye.aspx>, acedido em 20/11/2018

²⁴ **PT PC EYE Mini:** <http://www.fundacao.telecom.pt/Home/Acessoàscomunicações/SoluçõesEspeciaisPT/PTPCEYEMini.aspx>, acedido em 20/11/2018

Outra referência em Portugal no desenvolvimento e comercialização de produtos para a educação especial e pessoas com deficiências era a empresa CNOTINFOR – Centro de Novas Tecnologias da Informação, Lda., estava localizada na cidade de Coimbra. Esta empresa distribuía produtos direcionados para a educação especial e para as pessoas com deficiências, mas abriu insolvência e encerrou as suas atividades em maio de 2017, removendo todos os software das lojas de aplicações e websites. A seguir estão os produtos desenvolvidos por esta empresa:

Vox4All 2.0: Software com um sistema de comunicação alternativa e aumentativa, estava disponível nas plataformas Android, iOS e Windows, o seu preço era 69,99€. A figura 2.24 demonstra este software.



Figura 2.24: Vox4all

Oscar Feels: Aplicação para a aprendizagem de como gerir e lidar com as emoções, destinada principalmente para as crianças com autismo. Estava disponível apenas na plataforma Android e o preço era 149,99€. A figura 2.25 demonstra esta aplicação.



Figura 2.25: Oscar Feels

Symbol LAB: Aplicação para a criação e partilha de histórias adaptadas, fichas de trabalho e atividades interativas para comunicação e aprendizagem. Estava disponível apenas para os computadores com sistema operativo Windows e o preço era 232,30€. A figura 2.26 demonstra esta aplicação.

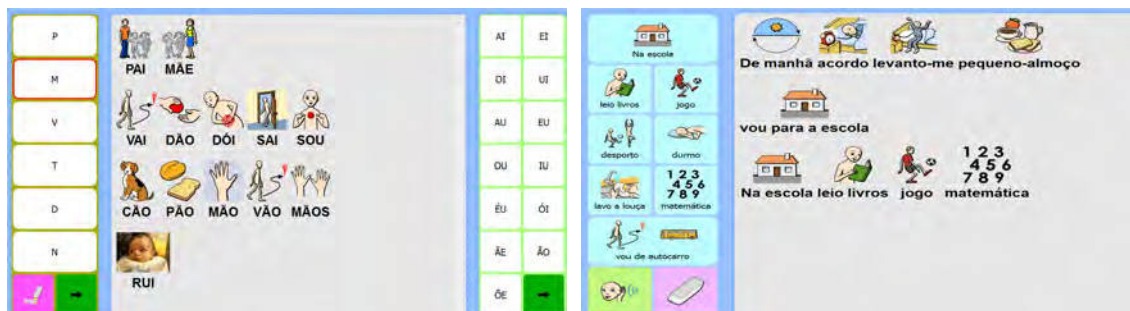


Figura 2.26: Symbol LAB

Em Portugal existem lojas especializadas em produtos destinados para as crianças na educação especial e as pessoas com deficiências. Foi realizada uma consulta de produtos em três das lojas com maior destaque nacional, sendo as empresas Anditec²⁵, MegaSerafim²⁶ e Oficina Didáctica²⁷, conforme é mostrado a seguir:

Look to Learn: Software para treino cognitivo e motor, é composto por 40 atividades para pessoas que estão a iniciar o uso de equipamentos como o rato, comandos, ecrã tátil, sensores de movimentos pelo olhar ou movimento da cabeça. Disponível apenas para os computadores com sistema operativo Windows, o preço é 444€ e 1.032€ com os comandos adaptados. A figura 2.27 demonstra este software.



Figura 2.27: Look to Learn²⁸

²⁵ **Anditec:** <http://www.anditec.pt>, acessado em 20/12/2018

²⁶ **MegaSerafim:** <https://www.megaserafim.pt/>, acessado em 20/12/2018

²⁷ **Oficina Didáctica:** <http://oficinadidactica.pt>, acessado em 20/12/2018

²⁸ **Look to Learn:** http://www.zyteq.com.au/products/software/look_to_learn, acessado em 20/12/2018

Mind Express 4: Software utilizado para comunicação alternativa e aumentativa com o SPC – Símbolos Pictográficos de Comunicação, disponível apenas para os computadores com sistema operativo Windows, o preço é 671€. A figura 2.28 demonstra este software.

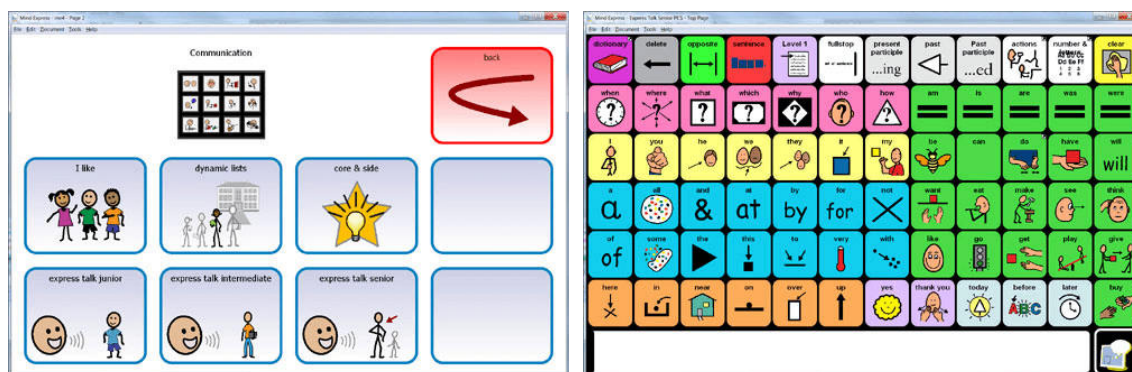


Figura 2.28: Mind Express 4²⁹

Comunicar com Símbolos: Software utilizado na comunicação alternativa e aumentativa com o SPC – Símbolos Pictográficos de Comunicação. É destinada para crianças em fase de pré-alfabetização e alunos na educação especial com dificuldade na escrita e de aprendizagem. Está disponível apenas para os computadores com sistema operativo Windows e o preço é 239,31€. A figura 2.29 demonstra este software.

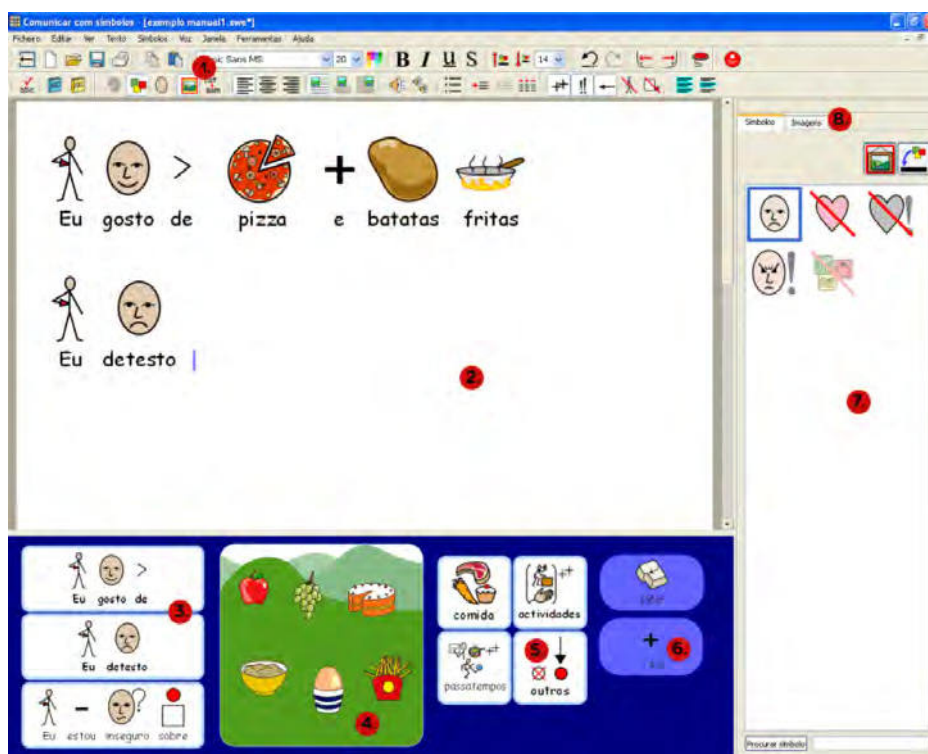


Figura 2.29: Comunicar com Símbolos³⁰

²⁹ **Mind Express 4:** http://www.zyteq.com.au/products/software/mind_express, acessado em 20/12/2018

³⁰ **Comunicar com Símbolos:** <http://oficinadidactica.pt/produto/comunicar-simbolos/>, acessado em 20/12/2018

Boardmaker & Speaking Dynamically Pro: São 2 software que se complementam, o Boardmaker contém mais de 5.000 símbolos pictográficos para a comunicação que são utilizados para criar quadros impressos. O Speaking Dynamically Pro permite associar um conjunto de ações a botões de quadros com o computador, disponível apenas para os computadores com sistema operativo Windows e o preço é 595€. A figura 2.30 demonstra este software.

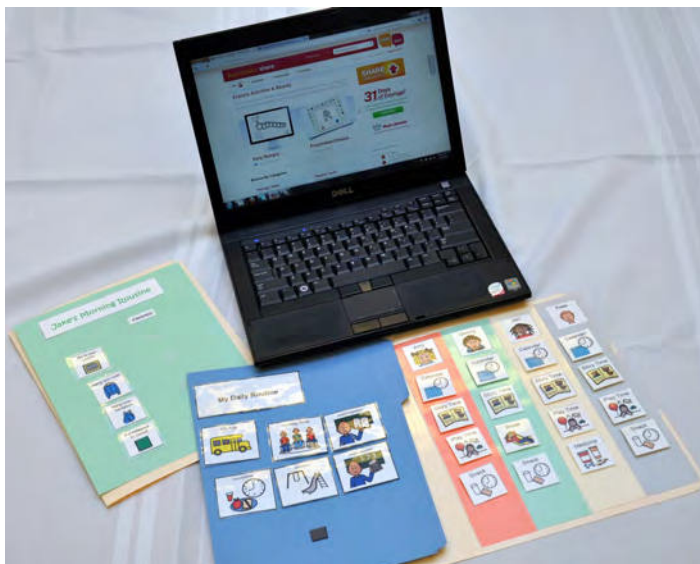


Figura 2.30: Boardmaker & Speaking Dynamically Pro ³¹

InVento 2: É um software utilizado para construir fichas para trabalhos escolares e livros impressos, com o suporte de 6.000 símbolos pictográficos e 1.500 imagens e fotografias coloridas. Está disponível apenas para os computadores com sistema operativo Windows e o preço é 149,91€. A figura 2.31 demonstra este software.

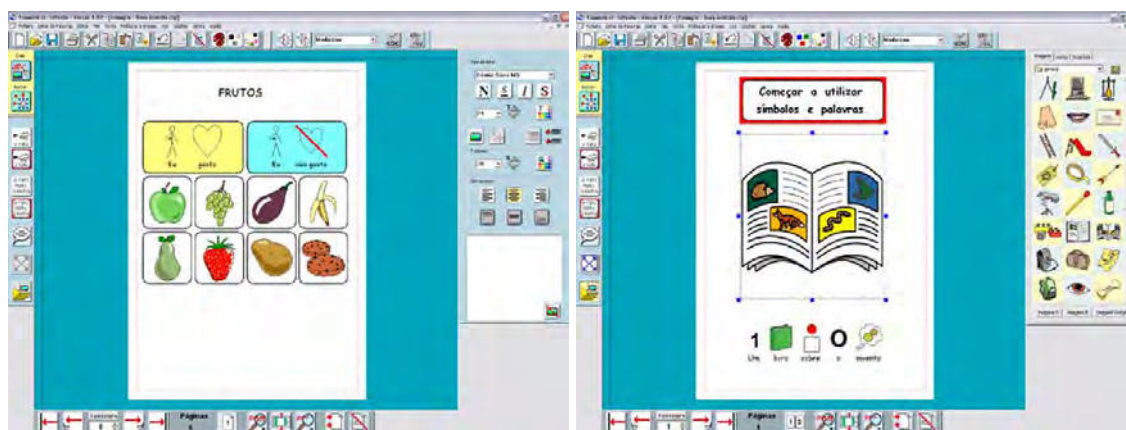


Figura 2.31: InVento 2 ³²

³¹ **Boardmaker & Speaking Dynamically Pro:** <https://www.megaserafim.pt/online/software-de-comunicacao/366-boardmaker-speaking-pro.html>, acedido em 20/12/2018

³² **InVento 2:** http://www.anditec.pt/index.php?option=com_virtuemart&view=productdetails&virtuemart_product_id=195&virtuemart_category_id=43, acedido em 20/12/2018

Jogos Memória – Junior: Software composto por jogos educativos desenvolvidos por psicólogos e educadores, utilizado no treino cognitivo para melhorar a memória. É vocacionado para crianças e adultos com dificuldade de concentração, está disponível para os computadores com sistema operativo Windows e o preço é 169€. A figura 2.32 demonstra este software.



Figura 2.32: Jogos Memória – Junior ³³

Letras e Palavras: É um software que possui uma coleção de atividades em formato de jogos para trabalhar o desenvolvimento da linguagem e literacia, permitindo que os profissionais (educadores, terapeutas, psicólogos, etc.) introduzam listas de palavras e frases. É destinado às crianças no ensino básico, também sendo utilizado para aprender outros idiomas, como o inglês. Está disponível para os computadores com sistema operativo Windows e o preço é 74,90€. A figura 2.33 demonstra este software.

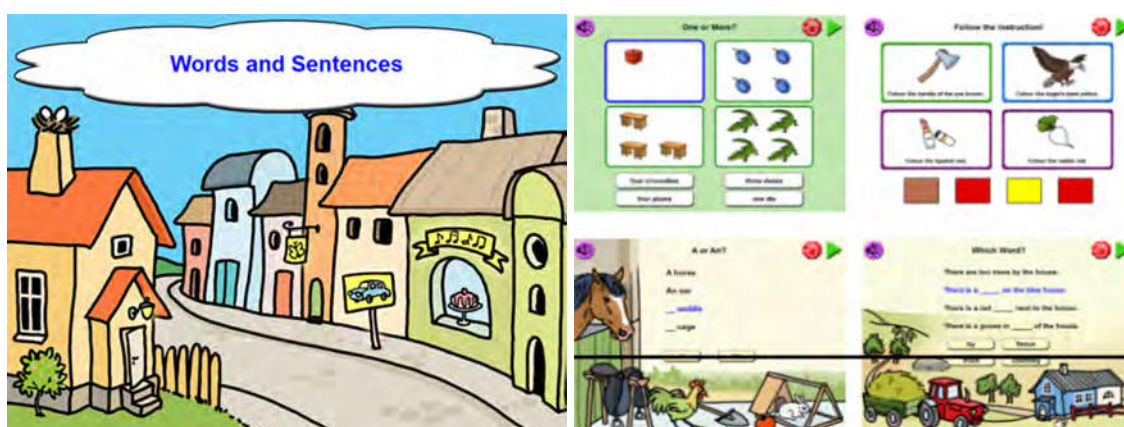


Figura 2.33: Letras e Palavras ³⁴

³³ **Jogos Memória:** <https://www.megaserafim.pt/online/multidisciplinar/140-software-jogos-memoria-junior.html>, acedido em 20/12/2018

³⁴ **Letras e Palavras:** <http://oficinadidactica.pt/produto/letras-e-palavras-21-atividades-para-a-linguagem-e-literacia/>, acedido em 20/12/2018

Primeiros Anos Mix: Software com jogos desenvolvidos por uma equipa de educadores, é utilizado especialmente na intervenção precoce em crianças com autismo ou dificuldade na linguagem e aprendizagem. Está disponível para computadores com sistema operativo Windows e o preço é 169€. A figura 2.34 demonstra este software.



Figura 2.34: Primeiros Anos Mix ³⁵

MIND7: É um software utilizado na intervenção direccionada para as competências de leitura e escrita, contém 50 atividades relacionadas com o quotidiano das pessoas e 30 exercícios para cada atividade. Está disponível para os computadores com sistema operativo Windows e o preço é 99,90€. A figura 2.35 demonstra este software.

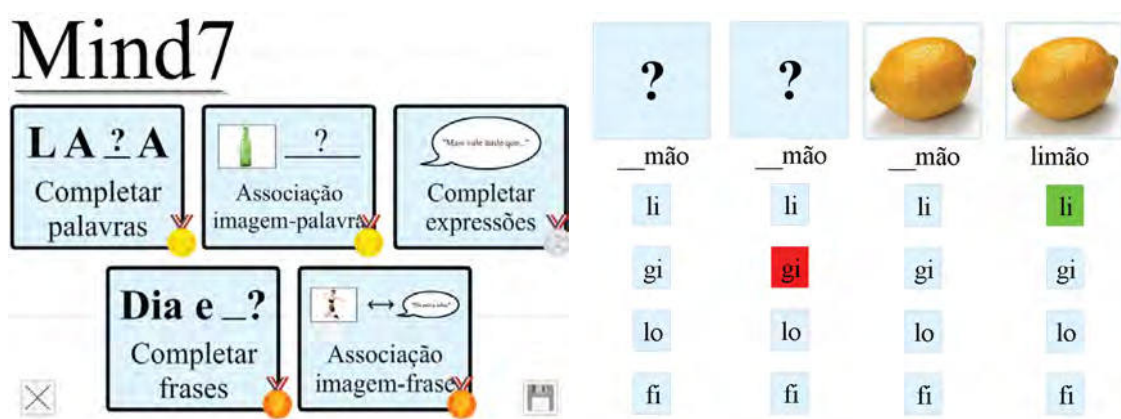


Figura 2.35: MIND7 ³⁶

³⁵ **Primeiros Anos Mix:** <https://www.megaserafim.pt/online/multidisciplinar/142-primeiros-anos-mix.html>, acedido em 20/12/2018

³⁶ **MIND7:** <http://oficinadidactica.pt/produto/mind7-intervencao-em-afasias/>, acedido em 20/12/2018

A seguir serão demonstradas aplicações móveis para este segmento, disponíveis para a plataforma Apple iOS, através da loja iTunes³⁷:

Proloquo2Go: Software que está disponível em diversos idiomas, fornece um sistema de comunicação alternativa e aumentativa. Tem o objetivo de auxiliar as pessoas com dificuldade na fala, fornecendo diversos recursos para facilitar a conversação e o preço é 249,99€. A figura 2.36 demonstra este software.

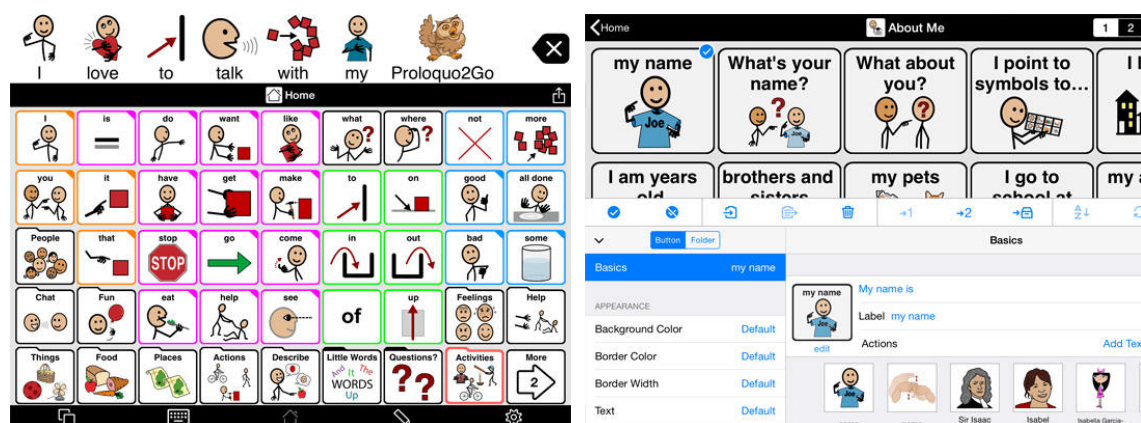


Figura 2.36: Proloquo2Go ³⁸

TouchChat HD: Software disponível nos idiomas inglês, espanhol, hebraico e árabe. É um sistema de comunicação alternativa e aumentativa desenvolvido para as pessoas com deficiência intelectual, pessoas com acidente vascular cerebral e outros problemas relacionados com a dificuldade na fala e o preço é 299,99€. A figura 2.37 demonstra este software.



Figura 2.37: TouchChat HD ³⁹

³⁷ iTunes: <https://itunes.apple.com>, acessado em 20/12/2018

³⁸ Proloquo2Go: <http://www.assistiveware.com/product/proloquo2go>, acessado em 20/12/2018

³⁹ TouchChat HD: <https://itunes.apple.com/us/app/touchchat-hd-aac-with-wordpower/id412351574>, acessado em 20/12/2018

Avaz Pro – AAC app for Autism: Software disponível em inglês, é uma aplicação para comunicação alternativa e aumentativa que utiliza os pictogramas para auxiliar as crianças que não falam ou tem dificuldade em falar, recebendo em 2011 diversos prémios internacionais. O preço é 199,99€. A figura 2.38 demonstra este software.

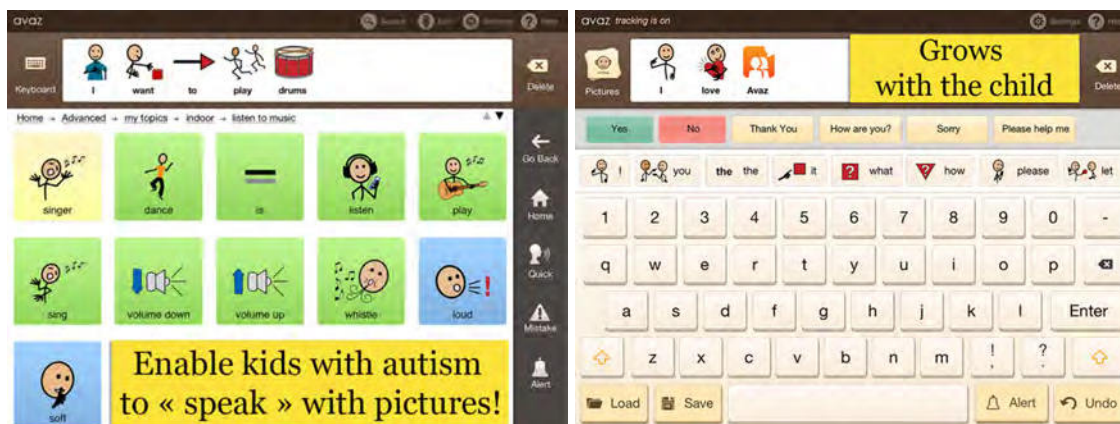


Figura 2.38: Avaz Pro – AAC app for Autism ⁴⁰

Social Skills With Billy: Software disponível em inglês, é uma aplicação para a terapia da fala, destinada para as crianças com autismo. O objetivo é desenvolver habilidades sociais e linguagem em situações do dia a dia. O preço é 14,99€. A figura 2.39 demonstra este software.



Figura 2.39: Social Skills With Billy ⁴¹

⁴⁰ **Avaz Pro – AAC app for Autism:** <https://itunes.apple.com/us/app/avaz-pro-aac-app-for-autism/id558161781>, acedido em 20/12/2018

⁴¹ **Social Skills With Billy:** <https://itunes.apple.com/us/app/social-skills-with-billy/id1027830019>, acedido em 20/12/2018

Autism/DTT Shapes: Software disponível nos idiomas inglês, mandarim tradicional e simplificado. É uma aplicação utilizada para ensinar formas geométricas, cores, letras, números, palavras, etc. É direcionado para as crianças com autismo, transtorno de déficit de atenção e hiperatividade ou crianças com problemas de concentração e aprendizagem. O preço é 7,99€. A figura 2.40 demonstra esta aplicação.

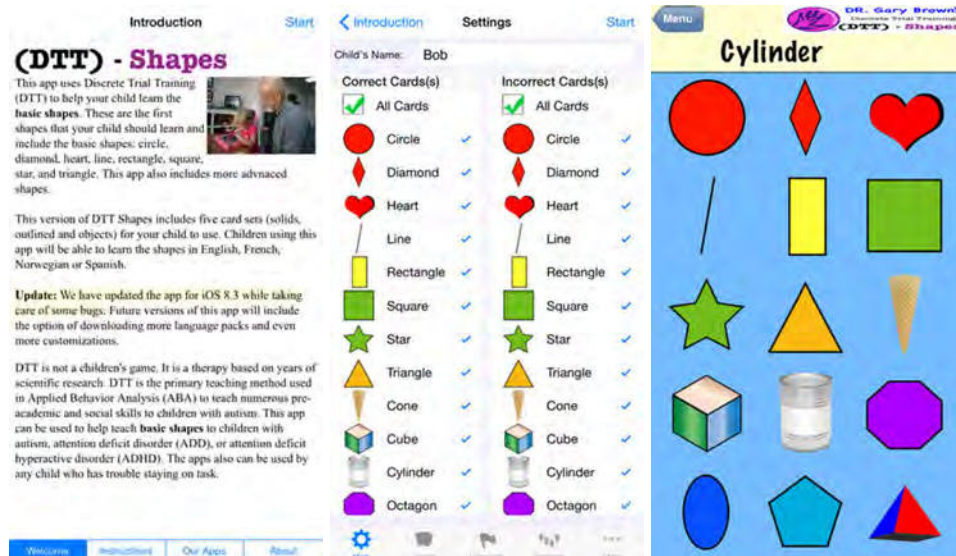


Figura 2.40: Autism/DTT Shapes ⁴²

iCommunicate: Software disponível em inglês, possui funcionalidades para a criação de quadros de comunicação, rotinas, cartões de memória e fala. É destinado para os alunos na educação especial e crianças em fase de pré-alfabetização e o preço é 49,99€. A figura 2.41 demonstra este software.

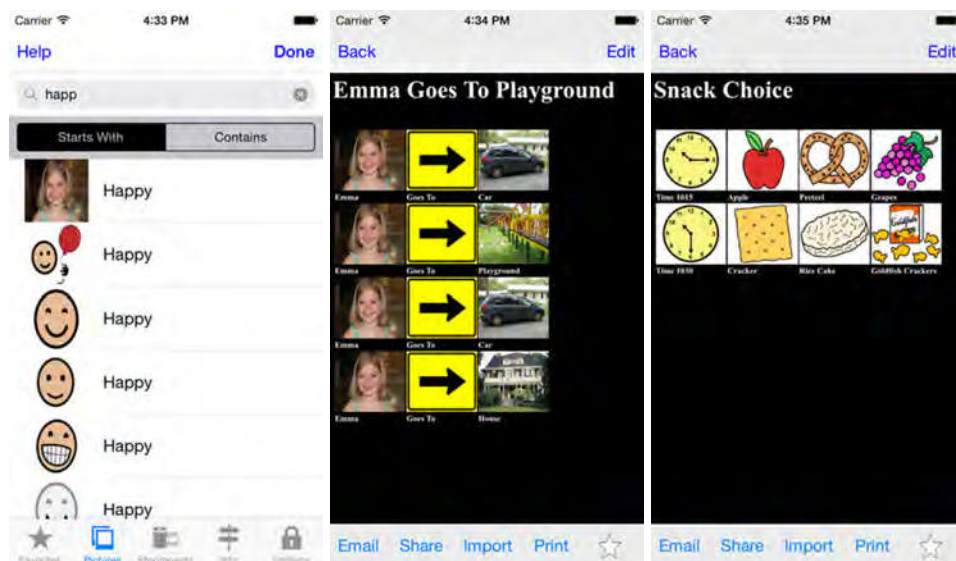


Figura 2.41: iCommunicate ⁴³

⁴² **Autism/DTT Shapes:** <https://itunes.apple.com/us/app/autism-dtt-shapes/id416380028>, acedido em 20/12/2018

⁴³ **iCommunicate:** <https://itunes.apple.com/us/app/communicate/id320986580>, acedido em 20/12/2018

iPrompts PRO: Software disponível em inglês, usado para ajudar os profissionais da saúde a criarem agendas, temporizadores e diversas atividades relacionadas com o autismo, o preço é 99,99€. A figura 2.42 demonstra este software.

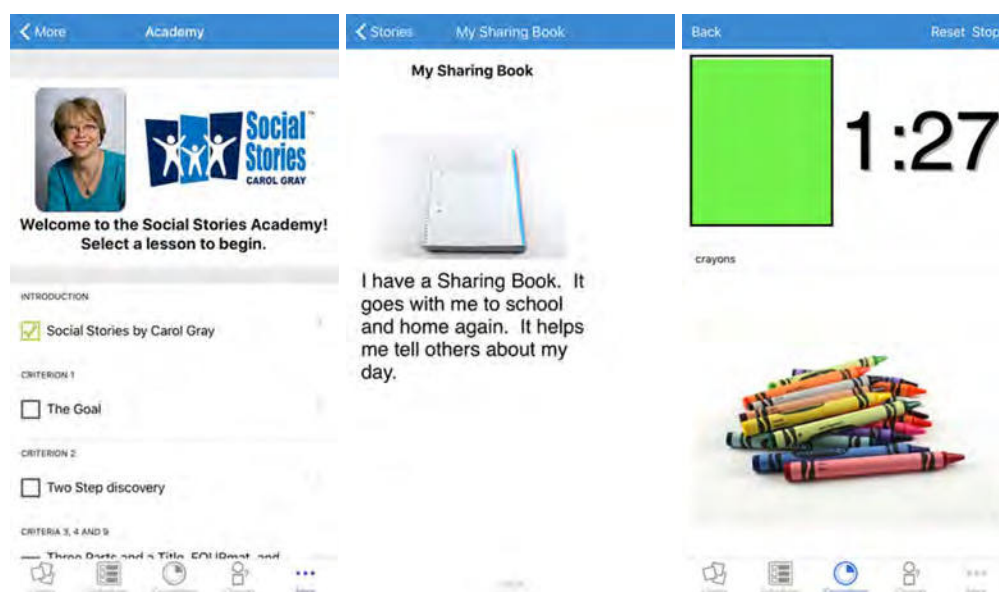


Figura 2.42: iPrompts PRO ⁴⁴

MyTalkTools: Software disponível no idioma inglês, contém três aplicações destinadas para as crianças com autismo e o preço é 99,99€. A figura 2.43 demonstra este software.

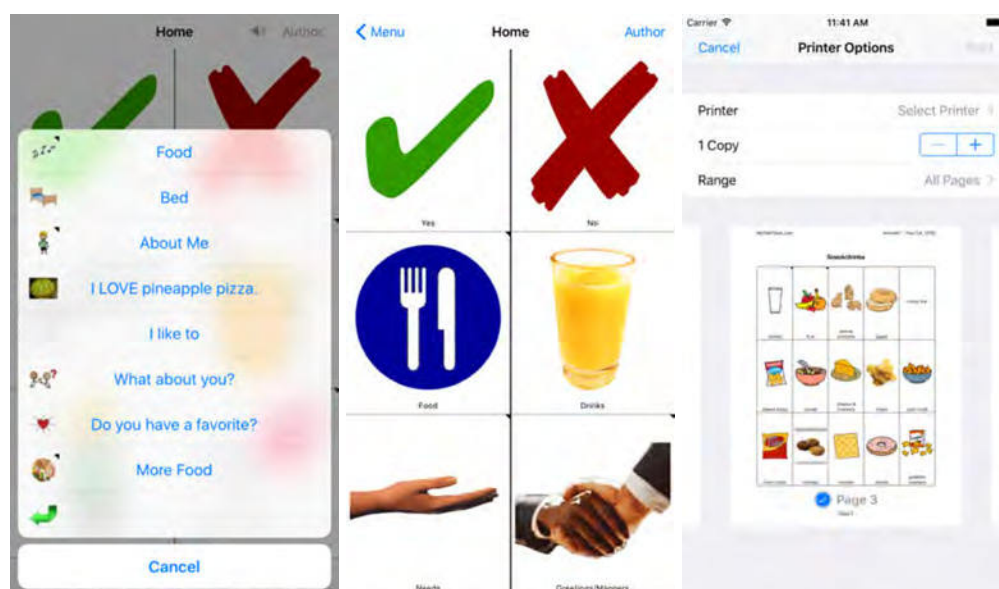


Figura 2.43: MyTalkTools ⁴⁵

⁴⁴ **iPrompts PRO:** <https://itunes.apple.com/us/app/iprompts-pro/id313144705>, acessado em 20/12/2018

⁴⁵ **MyTalkTools:** <https://itunes.apple.com/us/app/mytalktools-mobile/id324286288>, acessado em 20/12/2018

Abilipad: Software disponível em diversos idiomas como português, inglês, francês, alemão, espanhol, etc. Fornece um teclado customizado e a conversão de texto para voz e o preço é 19,99€. A figura 2.44 demonstra este software.

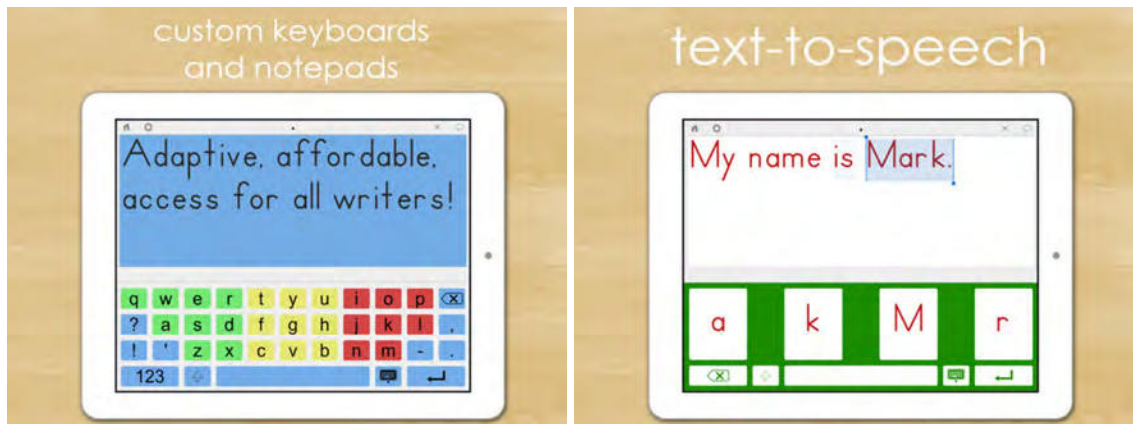


Figura 2.44: Abilipad ⁴⁶

Speech FlipBook Standard: Software disponível em inglês, fornece um conjunto de ferramentas para trabalhar a terapia da fala como a consciência fonológica, articulação, apraxia, leitura, etc. É destinado para todas as idades e o preço é 9,99€. A figura 2.45 demonstra este software.

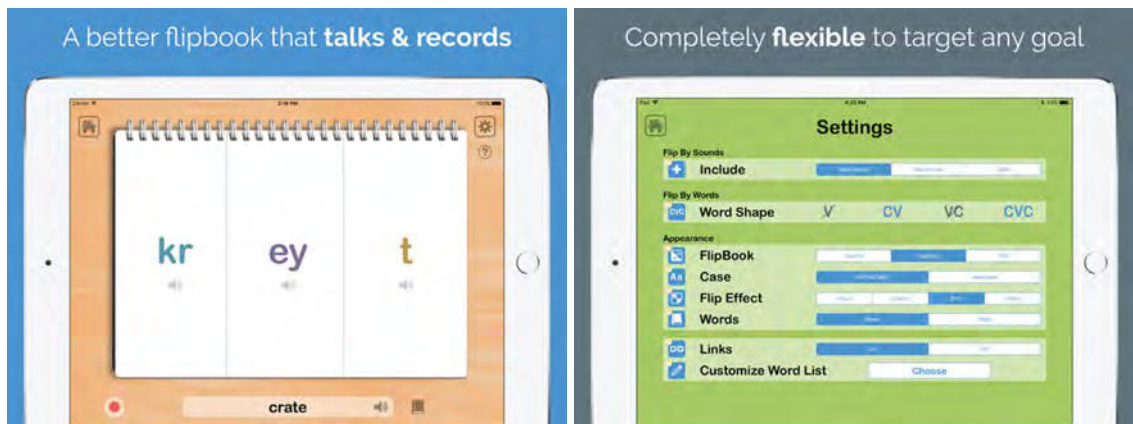


Figura 2.45: Speech FlipBook Standard ⁴⁷

⁴⁶ **Abilipad:** <https://itunes.apple.com/us/app/abilipad/id435865000>, acessado em 20/12/2018

⁴⁷ **Speech FlipBook Standard:** <https://itunes.apple.com/us/app/speech-flipbook-standard/id582842245>, acessado em 20/12/2018

A seguir serão descritas aplicações móveis neste segmento, disponíveis para a plataforma Google Android através da loja Play Store⁴⁸:

Voice4u AAC Communication: Software disponível em inglês, fornece um sistema de comunicação alternativa e aumentativa simples, é destinada para as pessoas com autismo e com problemas de saúde como acidente vascular cerebral, paralisia cerebral, etc. O preço é 9,90€ com produtos integrados para venda dentro da aplicação de 3,99€ até 29,99€ por item. A figura 2.46 demonstra este software.

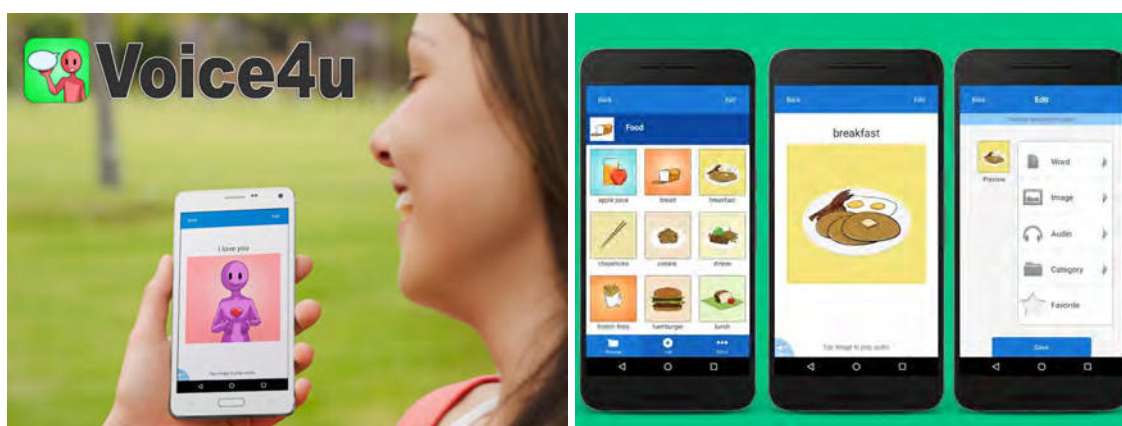


Figura 2.46: Voice4u AAC Communication ⁴⁹

LetMeTalk: Software disponível em diversos idiomas como inglês, espanhol, francês, italiano, português e alemão. É uma aplicação para a comunicação alternativa e aumentativa, fornece diversas funcionalidades para o apoio da comunicação. O preço é gratuito com venda dentro da aplicação de 2,49€ até 99,99€ por item. A figura 2.47 demonstra este software.

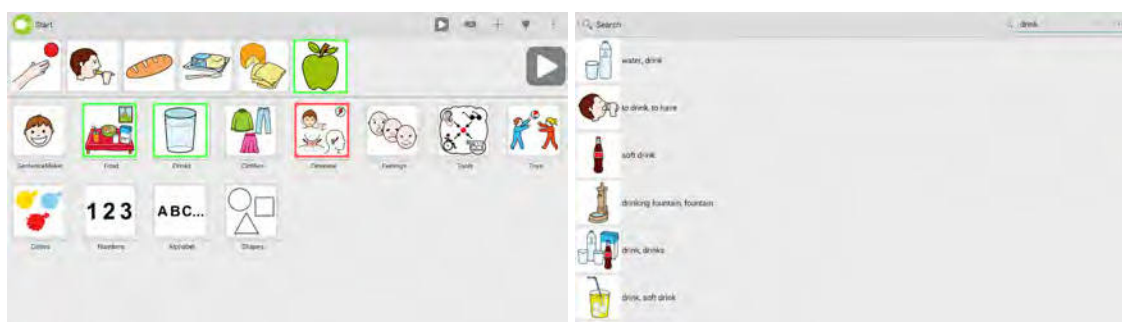


Figura 2.47: LetMeTalk ⁵⁰

⁴⁸ **Play Store:** <https://play.google.com>, acedido em 20/12/2018

⁴⁹ **Voice4u AAC:** <https://support.t-mobile.com/community/phones-tablets-devices/android/t-mobile-g1>, acedido em 20/11/2018

⁵⁰ **LetMeTalk:** <https://play.google.com/store/apps/details?id=de.appnotize.letmetalk>, acedido em 20/12/2018

TalkTablet: Software disponível em vários idiomas como inglês, espanhol, francês, italiano, português, alemão, etc. É uma aplicação para a comunicação alternativa e aumentativa com 30.000 pictogramas e a funcionalidade de converter o texto para voz, sendo direcionado para crianças na educação especial e pessoas com dificuldades na comunicação. O preço é 94,99€. A figura 2.48 demonstra este software.

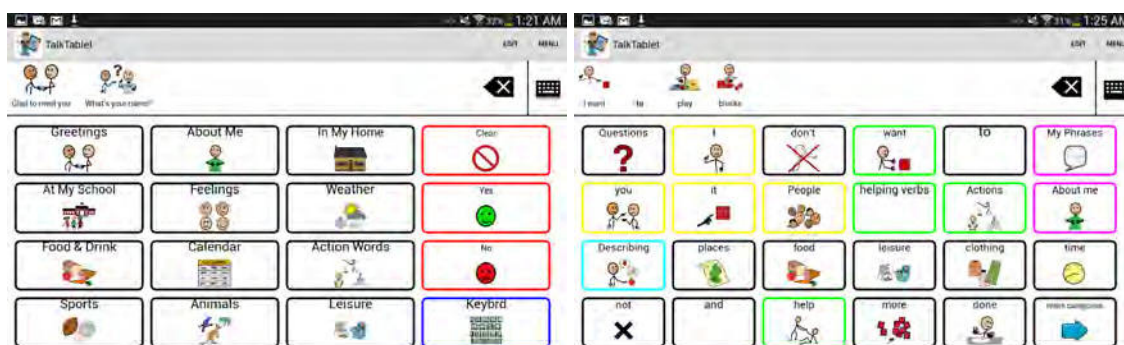


Figura 2.48: TalkTablet ⁵¹

Livox: Software disponível em diversos idiomas, é uma aplicação para a comunicação alternativa e aumentativa, foi eleita em 2014 pela Organização das Nações Unidas – ONU a melhor aplicação de inclusão social do mundo. Possui funcionalidades relacionadas com acessibilidade e mais de 24.000 pictogramas. É indicado para pessoas com autismo, trissomia 21, esclerose lateral amiotrófica, paralisia cerebral, sequelas de acidentes vasculares cerebrais, trauma crânio-encefálico e outras deficiências ou doenças que dificultem a comunicação oral. O preço é através de planos por mensalidades, sendo que o plano básico custa cerca de 99€ mensais. A figura 2.49 demonstra este software.



Figura 2.49: Livox ⁵²

⁵¹ **TalkTablet:** <https://play.google.com/store/apps/details?id=com.gusinc.talktableta>, acessado em 20/12/2018

⁵² **Livox:** <https://play.google.com/store/apps/details?id=br.com.livox>, acessado em 20/12/2018

Samsung LOOK AT ME: Software disponível gratuitamente em inglês, foi desenvolvido para as pessoas com autismo. O objetivo é melhorar o contacto visual e as suas capacidades de reconhecimento facial. Foi desenvolvido pela empresa Samsung em conjunto com profissionais como educadores, médicos, psicólogos e designers. A figura 2.50 demonstra este software.

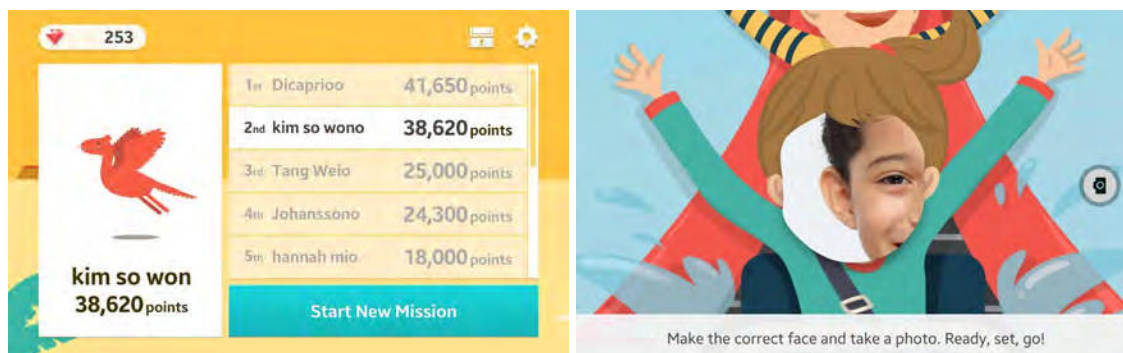


Figura 2.50: Samsung LOOK AT ME ⁵³

Montessori Words & Phonics: Software disponível no idioma inglês, foi desenvolvido com base no método de aprendizagem Montessori. O objetivo é ajudar as crianças desenvolverem as capacidades de leitura, escrita e ortografia. Contém um conjunto de 320 desafios que além de capacitar a literacia e fornece ajuda na consciência fonológica, o preço é 3,09€. A figura 2.51 demonstra este software.

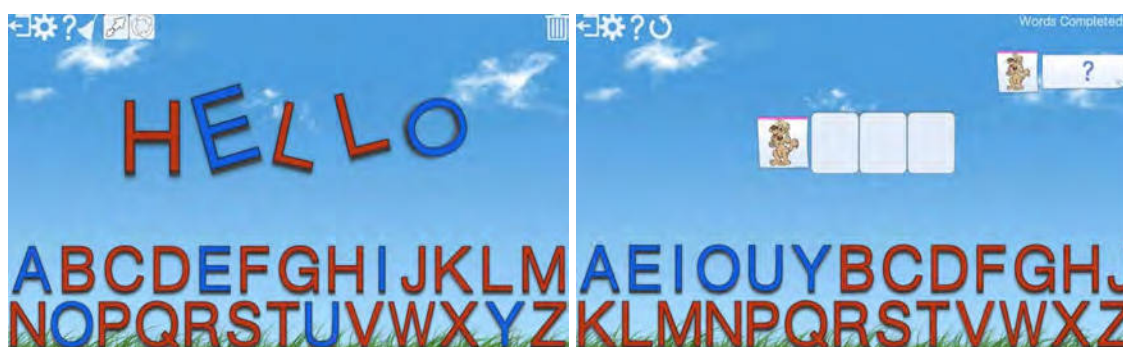


Figura 2.51: Montessori Words & Phonics ⁵⁴

⁵³ **Samsung LOOK AT ME:** <https://play.google.com/store/apps/details?id=com.samsung.lookatme>, acedido em 20/12/2018

⁵⁴ **Montessori Words & Phonics:** <https://play.google.com/store/apps/details?id=com.lescapadou.picturespelling>, acedido em 20/12/2018

Autism Language and Cognitive Therapy with MITA: Software está disponível em vários idiomas como inglês, espanhol, francês, italiano, etc., mas não está disponível em português. É uma aplicação direcionada para as crianças com autismo, possui um conjunto de puzzles que ajudam a melhorar o desenvolvimento cognitivo, principalmente da linguagem, atenção e habilidades visuais. O preço da versão básica é gratuito, sendo vendido cada item dentro da app por 12,99€. A figura 2.52 demonstra este software.

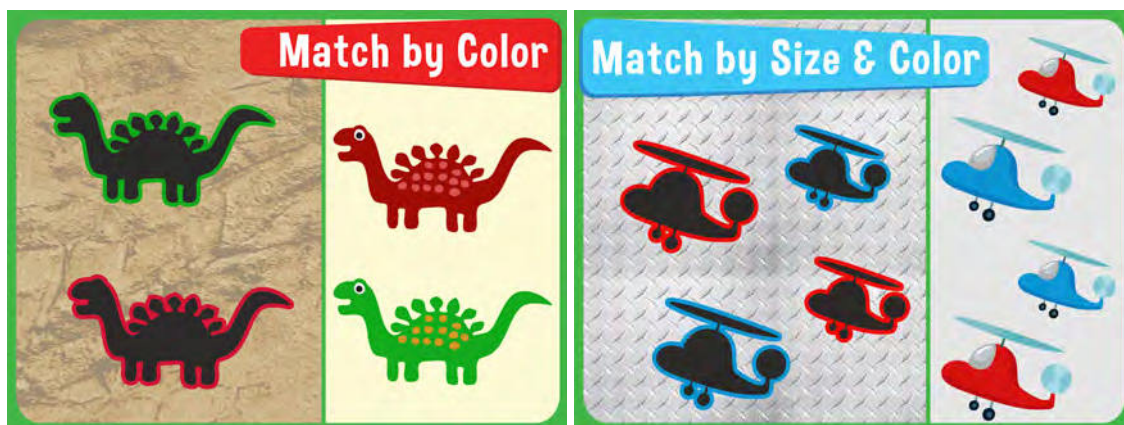


Figura 2.52: Autism Language and Cognitive Therapy with MITA ⁵⁵

EKUI Alfabeto: Software disponível em português, é uma aplicação utilizada como um material didático inclusivo para a alfabetização. Possui funcionalidades para trabalhar diversas áreas cognitivas e com traduções para braile e para a língua gestual portuguesa. O preço é 6,99€. A figura 2.53 demonstra este software.



Figura 2.53: EKUI Alfabeto ⁵⁶

⁵⁵ **Autism Language and Cognitive Therapy:** <https://play.google.com/store/apps/details?id=com.imagination.mita>, acedido em 20/12/2018

⁵⁶ **EKUI Alfabeto:** <https://play.google.com/store/apps/details?id=com.ekui.alfabetopro>, acedido em 20/12/2018

Puzzles educativos para miúdo: Software está disponível em diversos idiomas como português, inglês, espanhol, francês, italiano, etc., contém diversos jogos com o objetivo de melhorar o desenvolvimento cognitivo, como o pensamento crítico e raciocínio lógico, aprendizagem visual e percepção espacial, atenção, concentração, capacidades psicomotoras, etc. O preço é gratuito na versão básica, com vendas dentro da aplicação de 0,99€ até 3,09€ por item. A figura 2.54 demonstra este software.



Figura 2.54: Puzzles educativos para miúdo ⁵⁷

Smile and Learn: Jogos educativos para crianças: Software disponível nos idiomas português, inglês, espanhol, francês e italiano, contém mais de 100 atividades entre jogos e histórias infantis, é destinado para crianças entre 3 e 12 anos, tendo como objetivo a capacitação cognitiva e o entretenimento. Algumas das histórias infantis possuem a tradução com pictogramas e locução. O preço é gratuito da versão limitada, possui venda dentro da aplicação de 3,04€ até 4,09€ por item, além da opção de assinatura mensal de cerca de 3,53€. A figura 2.55 demonstra este software.



Figura 2.55: Smile and Learn: Jogos educativos para crianças ⁵⁸

⁵⁷ **Puzzles educativos para miúdo:** <https://play.google.com/store/apps/details?id=com.imagiRation.imagiRation>, acedido em 20/12/2018

⁵⁸ **Smile and Learn:** <https://play.google.com/store/apps/details?id=net.smileandlearn.library>, acedido em 20/12/2018

O CRID – Centro de Recursos para a Inclusão Digital⁵⁹, localizado na Escola Superior de Educação e Ciências Sociais do Instituto Politécnico de Leiria (ESECS/IPL), é um grande responsável pela promoção e apoio à comunidade na área da acessibilidade digital, direcionado principalmente para as pessoas com necessidades especiais de educação e para as pessoas com deficiências.

Uma vertente do CRID é o apoio e a criação de livros multiformato, permitindo que uma obra literária seja traduzida em diferentes formatos para a leitura, como texto, ilustrações, símbolos pictográficos de comunicação, impressão em braile e ilustrações em relevos, áudio com a locução e o vídeo com a língua gestual portuguesa. A seguir estão alguns exemplos de livros multiformato:

O menino dos dedos tristes: Publicado em 2012 pela editora Alfarroba com a autoria de Josélia Neves. É um livro com os símbolos pictográficos de comunicação, áudio-livro e vídeo com a tradução para língua gestual portuguesa. A figura 2.56 demonstra este livro.



Figura 2.56: O menino dos dedos tristes⁶⁰

⁵⁹ CRID: <http://crid.esecs.ipleiria.pt/>, acedido em 20/12/2018

⁶⁰ O menino dos dedos tristes: <http://omeninodosdedostristes.blogspot.com/>, acedido em 20/12/2018

Piu Caganita: Publicado em 2016 pela editora Briza, foi escrito e ilustrado por Tânia Bailão Lopes. Possui a impressão do texto em grande formato, pictogramas, braile e imagens em relevo, áudio-livro e a língua gestual portuguesa. A figura 2.57 demonstra este livro.



Figura 2.57: Piu Caganita ⁶¹

⁶¹ **Piu Caganita:** <https://bailaolopes.wordpress.com/projectos/piu-caganita/>, acedido em 20/12/2018

A Rainha das Rosas: Publicado em 2018, este livro foi desenvolvido pelo CRID e por 96 crianças das escolas da Freguesia de Cortes. Possui o suporte para texto aumentado, símbolos pictográficos de comunicação, braile, imagens impressas em relevo, áudio-livro e vídeo com a tradução para língua gestual portuguesa. A figura 2.58 demonstra este livro.



Figura 2.58: A Rainha das Rosas ⁶²

Este capítulo apresentou uma análise sobre a tecnologia na educação especial, mostrando informações sobre as NEE, TIC, dispositivos móveis e demonstrando um conjunto de produtos direcionados para a reabilitação das pessoas com deficiência e para os alunos na educação especial.

A relevância deste capítulo neste projeto foi investigar o cenário que envolve e a educação especial com uso da tecnologia. Estes temas são necessários para perceber o estado atual e facilitar a compreensão das necessidades nesta área, obtendo informações para fomentar o desenvolvimento de um software para este público.

⁶² **A Rainha das Rosas:** <https://apaiscortes.wixsite.com/arainhadasrosas/>, acedido em 20/12/2018

Capítulo 3

Caracterização de Deficiência

Neste capítulo são apresentadas informações sobre a deficiência, abordando a intelectual, auditiva e a surdez, uma vez que este projeto aplicado irá enfatizar estas deficiências na solução proposta. Demonstra as estatísticas das pessoas com deficiência e dos alunos na educação especial.

3.1 Deficiência

A Organização Mundial de Saúde estima que mais de um bilhão de pessoas possuem deficiência, sejam estas temporárias ou permanentes. Deste total, cerca de 200 milhões de pessoas com deficiência têm uma grande dificuldade funcional. Este número está a aumentar consideravelmente nos últimos anos, motivado pelo envelhecimento da população e pelo aumento das doenças crónicas (diabetes, problemas cardiovasculares, cancro, distúrbios mentais, etc.) [Organização Mundial da Saúde, 2012].

Em Portugal estima-se que existam cerca de 1 milhão de pessoas com deficiências (definitivas ou temporárias), mas este número pode ser superior. Existe um problema no recenseamento geral da população (CENSOS), não são utilizados os indicadores sobre as deficiências, utilizando os indicadores relacionados com as pessoas que têm dificuldades em executar as tarefas diárias.

A Associação Portuguesa de Deficientes – APD informou que em Portugal não se sabe ao certo a realidade das pessoas com deficiência. Os últimos CENSOS omitem informações sobre as pessoas com deficiências. A secretária de Estado para a Inclusão das Pessoas com Deficiência pretende realizar um levantamento do total de pessoas com deficiências, mas a data não foi estabelecida até o momento.

As pessoas com deficiências tendem a apresentar uma pior perspectiva em relação à saúde, além de outros fatores como baixos níveis de escolaridade, carência econômica e alta taxa de pobreza. A maior parte destas pessoas não possuem acesso adequado à saúde, à educação, tem menos oportunidades de emprego e sofrem de alguma forma um tipo de exclusão social [Organização Mundial da Saúde, 2012].

Nos últimos anos surgiram várias medidas para alterar este cenário, principalmente através da preparação da sociedade para lidar com as pessoas com deficiência, criando novas soluções para ultrapassar as barreiras nesta área, olhando para estes problemas como um assunto dos direitos humanos e não apenas como uma questão de saúde.

A deficiência compreende uma interação entre problemas de saúde e fatores contextuais, tanto pessoais quanto ambientais. Excluindo a questão da saúde, o que torna uma pessoa incapacitada geralmente é o ambiente e a falta de recursos, não sendo apenas uma limitação física ou intelectual [Organização Mundial da Saúde, 2012].

A terminologia sobre deficiência definida no artigo 2.º da Lei n.º 38/2004, de 18 de agosto [Diário da República, 2004]:

“Considera-se uma pessoa com deficiência aquela que, por motivo de perda ou anomalia, congénita ou adquirida, de funções ou de estruturas do corpo, incluindo as funções psicológicas, apresente dificuldades específicas suscetíveis de, em conjugação com os fatores do meio, lhe limitar ou dificultar a atividade e a participação em condições de igualdade com as demais pessoas.”

No último recenseamento geral da população (CENSOS) em 2011, houve uma mudança nos critérios de pesquisa. Foi substituída a terminologia “pessoa com deficiência” por “pessoa com dificuldade na realização de atividades do quotidiano devido a problemas de saúde ou decorrentes da idade”. Estas propostas foram elaboradas pelo “Washington Group on Disability Statistics”, sendo que os novos indicadores adotados têm o objetivo de avaliar e medir o grau de dificuldade e/ou incapacidade das pessoas, de acordo com:

- Dificuldade em ouvir, mesmo usando aparelho auditivo.
- Dificuldade em andar ou subir degraus.
- Dificuldade de memória ou de concentração.
- Dificuldade em tomar banho ou vestir-se sozinho.
- Dificuldade em compreender os outros ou fazer-se entender.

As novas classificações adotadas são divididas em 3 graus de dificuldade:

- Não tem dificuldade ou possui pouca.
- Tem muita dificuldade.
- Não consegue.

A tabela 3.1 demonstra o total de pessoas com dificuldade na realização das atividades diárias e a figura 3.1 apresenta o percentual deste total [GEP, 2016]:

Tabela 3.1: Total da população portuguesa com dificuldade nas atividades diárias

Tipo de dificuldade	Total de pessoas com dificuldade
Ver	399.899
Ouvir	471.557
Andar	655.377
Memória ou Concentração	980.000
Tomar banho ou vestir-se	523.202
Compreender ou fazer-se entender	920.519
Total da população com dificuldades	3.950.554

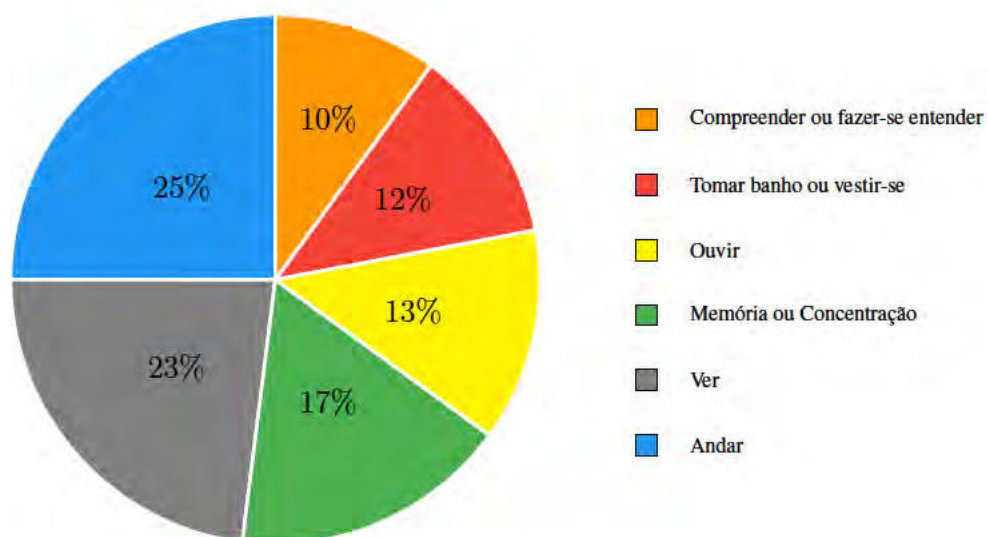


Figura 3.1: Percentual da população portuguesa com dificuldade em realizar alguma das atividades diárias

Em relação aos alunos com necessidades especiais de educação que frequentam as escolas regulares de ensino, os resultados no ano letivo 2015/2016 estão a seguir na tabela 3.2 [GEP, 2016]:

Tabela 3.2: Alunos com necessidades especiais de educação ano letivo 2015/2016

Nível	Alunos
Educação pré-escolar	3.573
Ensino básico	63.540
Ensino secundário	11.062
Total	78.175

A seguir estão algumas informações relevantes sobre o “Questionário das Necessidades Especiais de Educação Ano letivo 2015/2016” [DGEEC, 2016]:

- O total de alunos com necessidades especiais de educação foi 4% superior em comparação ao ano letivo anterior, 2014/2015.
- O total de alunos que frequentam as unidades de apoio especializado aumentou em relação ao ano letivo anterior, 2014/2015, quer nas unidades de apoio especializado para a educação de alunos com multideficiência e surdo-cegueira congénita (de 2.003 para 2.081 alunos, acréscimo de 3,9%), quer nas unidades de ensino estruturado para a educação de alunos com perturbações do espectro do autismo (de 1.699 para 1.878 alunos, acréscimo 10,5%).
- Houve uma redução de 5,2% no número de crianças apoiadas pelo Sistema Nacional de Intervenção Precoce na Infância.
- Do total de alunos com necessidades especiais de educação apenas 13% não estão a tempo integral na sua turma.
- O número total de professores que desempenham funções de educação especial nas escolas públicas aumentou em 12,7% (de 6.031 para 6.797)
- Em relação ao total dos técnicos nas escolas públicas com funções de apoio para os alunos com necessidades especiais de educação, o total diminuiu em 10,4% (de 931 para 834 técnicos) e o número de horas mensais associadas ao desempenho de funções aumentou quase na mesma proporção – 9,6% (de 56.464 para 61.897 horas).

- Em relação ao papel desempenhado pelos Centros de Recurso para a Inclusão, o número de agrupamentos apoiados pelos centros aumentou de 574 para 580 e o total de técnicos disponibilizado pelos centros aumentou em 26,1% (de 2.236 para 2.819).
- O número de Agrupamento de Referência para a Intervenção Precoce manteve-se o mesmo em relação ao ano letivo anterior.
- O número de docentes a desempenharem funções na área da intervenção precoce diminuiu em 12,2% (de 419 para 368).

3.2 Deficiência Intelectual

O termo deficiência intelectual foi utilizado oficialmente em 1995, durante o simpósio “Intellectual Disability: Programs, Policies and Planning for The Future”. Foi um substituto para o termo deficiência mental, usado incorretamente durante muitos anos para descrever a deficiência intelectual. A deficiência mental está relacionada com a área da psiquiatria e não com a área cognitiva e intelectual [Sasaki, 2005].

A adoção por completo do termo deficiência intelectual ocorreu durante a conferência considerada uma das mais importantes nesta área, onde foram discutidas e elaboradas mudanças necessárias para o desenvolvimento de uma educação mais inclusiva, originando a “Declaração de Salamanca e Enquadramento da Ação na Área das Necessidades Especiais de Educação” [UNESCO, 1994].

A deficiência intelectual é caracterizada por limitações no desenvolvimento cognitivo (raciocínio, aprendizagem e resolução de problemas) e dificuldades de comportamento adaptativo (comunicação, competências sociais, sentido de responsabilidade, atividades do quotidiano, aptidões académicas e profissionais) [França, 2008].

As causas da deficiência intelectual são incertas, existem fatores de risco que elevam a probabilidade da sua ocorrência, porém não devem ser consideradas responsáveis pela sua origem. De acordo com Pestana (2011) alguns dos principais fatores de risco são [Pestana, 2011]:

- **Condições genéticas:** Genes anormais herdados dos pais ou por outras razões de natureza genética.

- **Gestação:** Desenvolvimento do embrião ou feto, problemas na divisão das células, além de outros fatores como o tabagismo, alcoolismo, consumo de drogas, efeitos colaterais de medicamentos, desnutrição materna, doenças maternas causadas por infecções.
- **Nascimento:** Problemas durante o parto, como oxigenação cerebral insuficiente, prematuridade ou baixo peso.
- **Saúde:** Doenças como sarampo e meningite podem ter relação com a origem da deficiência intelectual. Além de problemas como má alimentação, contacto com medicamentos ou substâncias tóxicas como inseticidas, produtos químicos como chumbo, mercúrio, acidentes de viação, choque elétrico, afogamento, asfixia, etc.

As perturbações mais comuns associadas com a deficiência intelectual, ainda segundo o mesmo autor são:

- **Autismo:** É uma perturbação global do desenvolvimento infantil, que se prolonga por toda a vida e evolui com a idade. Estima-se que 1 em cada 160 crianças têm autismo. O espectro do autismo é um transtorno caracterizado por alterações em diferentes áreas do neurodesenvolvimento, principalmente no desenvolvimento cognitivo, sensorial, linguístico e social [Sousa e dos Santos, 2005].
- **Asperger:** É uma perturbação neurocomportamental de origem genética, manifestando através de alterações na interação social, comunicação e comportamento, pode ser considerada uma forma de autismo, conforme a informação disponibilizada pela “Associação Portuguesa do Síndrome de Asperger”¹.
- **Trissomia 21:** É um síndrome causado por um distúrbio genético no par de cromossomas 21 durante a divisão celular do embrião em seu desenvolvimento, sendo que a célula possui como informação genética 47 cromossomas em vez de 46 [Quelhas e Mesquita, 2011]
- **Síndrome de X Frágil:** É causado por uma mutação genética que impossibilita a produção da proteína FMRP (Fragile X Mental Retardation Protein). Os fatores hereditários são considerados como os principais responsáveis desta perturbação [Azevedo, 2011].

¹ O que é a Síndrome de Asperger?: <https://www.apsa.org.pt/pt/sindrome-de-asperger/o-que-e>, acedido em 12/11/2018

- **Síndrome de Prader-Willi:** É causado por um problema genético em que ocorre uma anomalia cromossômica que implica na ausência de alguns genes no cromossoma 15. É provocada pela falta de informação genética do pai, existindo informação genética apenas relativa à mãe [Kuo et al., 2007].
- **Síndrome de Angelman:** É considerado um síndrome irmão da Prader-Willi, também é causado por um problema genético, ocorrendo uma anomalia cromossômica muito similar a de Prader-Willi com a diferença que a informação genética em falta é a da mãe, havendo apenas informação genética do pai no cromossoma 15 [Maris e Trott, 2011].
- **Síndrome de Williams:** É causada por uma alteração de ordem genética através de uma anomalia cromossômica que implica a perda de 21 genes no cromossoma 7. É considerada muito rara e frequentemente não é diagnosticada com facilidade [Almeida e Tavares, 2010].

3.3 Deficiência Auditiva e Surdez

Existe uma diferenciação entre deficiência auditiva e surdez, embora geralmente a maioria das pessoas tenham dificuldade em compreender e distinguir.

A deficiência auditiva é um déficit adquirido, implica na redução da capacidade de ouvir, provocando uma audição insuficiente para as tarefas diárias. É caracterizada por dois tipos, uma deficiência temporária (com possibilidade de ultrapassar este problema através de procedimentos médicos) e uma definitiva (sem alternativas médicas para superar este déficit) [Feijão, 2015].

A surdez é um problema de origem congênita, é caracterizada como uma deficiência sensorial que leva à ausência total ou parcial da audição. Pode ser originada por fatores ambientais, genéticos e na combinação de ambos [Assunção, 2014]:

- **Fatores ambientais:** Pode estar associada com infecções pré e pós-natais, sendo que durante a gestação a progenitora contrai doenças que são originadas por vírus (herpes, rubéola e citomegalovírus) que comprometem o desenvolvimento do feto ou embrião. Outros fatores que podem ser considerados como as causas podem ser a prematuridade ou baixo peso, efeitos colaterais de medicamentos, exposição elevada a ruídos e traumas.

- **Fatores genéticos:** Podem estar associados com síndromes de ordem genética como a Síndrome de Usher, Síndrome de Pendred, Síndrome Jervell e Lange-Nielsen, Síndrome Alport e a Síndrome Norrie. A maioria dos casos de surdez de ordem genética são relacionados com os problemas hereditários, sendo a falta de audição a única patologia presente.
- **Fatores ambientais e genéticos:** Por exemplo, quando ocorre a perda ou redução da audição devido a uma anomalia genética (uma mutação no ADN mitocondrial) e o consumo de medicamentos ototóxicos (antibióticos da família dos aminoglicosídeos), a origem da surdez pode ser considerada como uma interação de fatores ambientais e genéticos.

Ainda de acordo com o Assunção (2014) a surdez pode ser de 3 tipos [Assunção, 2014]:

- **Condução:** Alterações no ouvido externo e/ou médio que limita ou impede a passagem do som no canal auditivo.
- **Neurosensorial:** Alterações do ouvido interno, quando a cóclea ou os condutores nervosos sofrem degeneração que impede os estímulos auditivos de chegarem até o cérebro.
- **Mista:** É resultado da junção de uma surdez neurosensorial com a condutiva.

Capítulo 4

Desenvolvimento da Aplicação “O menino que tinha medo do escuro”

Neste capítulo é abordado o desenvolvimento da solução proposta, detalhando o game design, a elaboração dos requisitos, a arquitetura de software e a plataforma Android. Demonstra como foi o processo de desenvolvimento da aplicação “O menino que tinha medo do escuro”.

4.1 Solução Desenvolvida

A proposta foi desenvolver uma aplicação móvel composta por um livro multimédia inclusivo e um conjunto de jogos lúdico-pedagógicos.

Foram realizadas algumas parcerias com profissionais e instituições para o desenvolvimento desta solução, entre elas estão:

- Susana Campos (autora) – Permitiu a utilização da história infantil “O menino que tinha medo do escuro”.
- Rui Lobo (ilustrador) – Concedeu o direito para reutilizar as ilustrações do livro infantil “O menino que tinha medo do escuro”.
- Centro de Recursos para Inclusão digital (CRID) – Forneceu a adaptação da história com os símbolos pictográficos de comunicação e o vídeo traduzido para a língua gestual portuguesa.
- Célia Sousa¹ – Ajudou na orientação do projeto, através da sua vasta experiência nesta área, é doutora em Ciências da Educação, especialista na formação de professores, comunicação aumentativa e não verbal, tecnologias de apoio, educação especial e problemas graves de cognição. É coordenadora do CRID com projetos nas áreas das tecnologias de apoio e da comunicação aumentativa.

¹ Célia Sousa: <http://iact.ipleiria.pt/member/celia-sousa/>, acedido em 29/04/2019

A aplicação fornece três modelos de livro multimédia, apresentando a história com os seguintes modos de leitura e interpretação:

- **Sem adaptações:** Os conteúdos são apresentados através de texto, áudio com a locução da história e a ilustração.
- **Adaptada com os pictogramas:** Utilizam texto, ilustração e símbolo pictográfico de comunicação. Foi adicionada a funcionalidade de converter o texto para a voz nos pictogramas, através da síntese de fala dos dispositivos.
- **Tradução para LGP:** Apresenta a história através de um vídeo com a tradução para língua gestual portuguesa e com a ilustração.

Em relação aos jogos lúdico-pedagógicos, foi realizada uma parceria com a educadora e especialista em educação especial Fernanda Bastos, que forneceu as orientações para elaborar os jogos nas áreas cognitivas, como:

Memória

A memória consiste no sistema de armazenamento e recuperação, concebido através de estímulos sensoriais com dois componentes diferentes e interligados, um componente é a estrutura que se estende à forma de como o sistema mnemónico (técnicas de memorização) está organizado e o outro componente é o processo de funcionamento às atividades que ocorrem dentro do sistema de memória [Oliveira, 2013].

A solução desenvolvida para trabalhar a memória foi um jogo de cartas com o objetivo de formar os pares da mesma figura e que estão distribuídos em posições aleatórias. As figuras utilizadas nas cartas são elementos das ilustrações e os símbolos pictográficos de comunicação. Existem 6 níveis de dificuldade de acordo com o número de cartas e com a variação do tempo que ficam visíveis ao iniciar o jogo, além da forma de visualização no início (uma carta por vez ou todas ao mesmo tempo).

Processamento psicomotor

O processamento psicomotor ou psicomotricidade é definido como a correlação intrínseca e indissociável entre a mente (psico) e corpo (motricidade). Associa-se com o movimento, mas como um meio, um suporte que auxilia as pessoas a adquirir o conhecimento do mundo que as rodeiam [Oliveira, 2013].

A solução desenvolvida para trabalhar os aspetos do processamento psicomotor foi um quebra-cabeças, com o objetivo de montar a imagem original, movimentando as peças para o local correto.

Imaginação, Criatividade e Desenho

Segundo o psicólogo bielorrusso Vygotsky² a imaginação é uma atividade combinatória cerebral, pertencente ao campo do irreal, sem valor prático, porém, funcionando como a base de toda atividade criativa, artística, científica e técnica. A imaginação é a combinação de experiências entre o sentimento, o pensamento e a realidade [Oliveira, 2013].

A criatividade pode ser considerada uma forma de pensamento divergente, caracterizado pela flexibilidade, fluidez e pela originalidade do pensamento [Oliveira, 2013].

O desenho é uma atividade de expressão plástica usual e natural, geralmente praticada individualmente. Pode ser utilizado pelas crianças como uma forma criativa e imaginativa para se divertirem [Oliveira, 2013].

A solução desenvolvida para trabalhar com a criatividade e imaginação foi um jogo de desenhar, com as opções de cores e formas geométricas, sem nenhuma restrição para o utilizador.

Linguagem e Ação

As pessoas com défice cognitivo podem apresentar dificuldades em relação ao domínio da leitura e da escrita, sendo que isto tem impacto no desenvolvimento da inteligência verbal e linguística que são utilizadas no processamento da linguagem [Oliveira, 2013].

A solução desenvolvida pretende trabalhar o vocabulário, memória, aptidão perceptiva, visual e auditiva, além da coordenação motora.

O objetivo deste jogo é fazer com que o utilizador complete com letras uma determinada palavra, fornecendo a funcionalidade para ouvir a pronúncia através da conversão do texto para voz utilizando a síntese da fala, além de visualizar a figura que representa a palavra.

4.2 Processo de Desenvolvimento de Software

Esta secção detalha como foi o desenvolvimento da aplicação, está dividida em cinco partes “4.2.1 Requisitos”, “4.2.2 Game Design”, “4.2.3 Arquitetura de Software”, “4.2.4 Plataforma Android” e “4.2.5 Implementação da Aplicação”. Abordam os processos da engenharia de software, desde o levantamento de requisitos até a implementação e testes da aplicação.

²Lev Vygotsky (1896-1934) foi um dos maiores pesquisadores sobre o desenvolvimento intelectual das crianças, é uma das principais referências neste tema até os dias atuais.

4.2.1 Requisitos

Seguindo o exemplo do documento de Game Design Canvas que será abordado na próxima secção, optou-se por desenvolver um conjunto simplificado de requisitos, com o objetivo de facilitar o processo de análise e desenvolvimento das funcionalidades.

A seguir estão os requisitos funcionais e não funcionais elaborados no desenvolvimento desta aplicação:

Funcionais

[RF01] – Seleção dos módulos

O utilizador deverá possuir opções no menu principal para selecionar qualquer um dos módulos na aplicação. Existirão cinco opções para escolha, três modos para a leitura e interpretação do livro multimédia e quatro jogos lúdico-pedagógicos.

[RF02] – Ativar e desativar o áudio

Na barra superior da aplicação haverá uma opção para ativar e desativar o áudio (música, efeitos sonoros e locução) em todos os módulos (livros multimédias e jogos). Esta opção deverá ser guardada na memória interna do dispositivo, para recuperar o valor da última escolha do utilizador na aplicação.

[RF03] – Opção para voltar ao menu principal e para encerrar a aplicação

Na barra superior da aplicação deverá existir a opção para sair se o utilizador estiver no menu principal e de voltar para o menu principal quando estiver nos módulos. Esta funcionalidade é a mesma do botão físico “Voltar” do dispositivo. O utilizador será alertado através de uma caixa de diálogo para confirmar a ação de sair da aplicação ou voltar para o menu principal.

[RF04] – Opção de informações complementares sobre a aplicação

Na barra superior da aplicação, dentro do menu principal deverá existir a opção de informações, com o objetivo de apresentar detalhes sobre o projeto. Este conteúdo será apresentado para o utilizador através de uma caixa de diálogo.

[RF05] – Livro multimédia básico

Será o livro multimédia de leitura e interpretação da história básica, irá utilizar conteúdos audiovisuais como texto e ilustração, além do áudio com a locução da história. A movimentação das páginas será com o movimento de *swipe* (deslizar do dedo no ecrã) e com botões para avançar e retroceder localizados na barra superior da aplicação.

[RF06] – Livro multimédia com pictogramas

Será o módulo do livro multimédia complementado com os símbolos pictográficos de comunicação. Os pictogramas deverão ter a funcionalidade de converter o texto em voz com a síntese da fala ao selecionar um elemento pictográfico. A movimentação entre as páginas e o campo com os pictogramas será realizada através do movimento de *swipe* e *scroll* (deslizar do dedo no ecrã).

[RF07] – Livro multimédia com a tradução para a língua gestual portuguesa

Será o módulo do livro multimédia com a leitura e interpretação da história através do vídeo com a tradução para língua gestual portuguesa. Este módulo não contém o texto e a locução da história. A movimentação entre as páginas será com o movimento de *swipe* (deslizar do dedo no ecrã) e com os botões para avançar e retroceder, opção para tocar e pausar o vídeo com a língua gestual portuguesa.

[RF08] – Jogo de desenhar

Será o módulo do jogo com pintura e desenho, contém as opções para escolha de cores, formas geométricas e tamanho do ponto ao tocar no ecrã. A pintura ou desenho será efetuada com o toque e a movimentação no ecrã, deverá possuir opções para apagar o ecrã todo, selecionar as cores e formas geométricas.

[RF09] – Jogo de memória

Será o módulo do jogo de memória, irá funcionar com a seleção dos pares de cartas com as mesmas figuras. A quantidade das cartas serão de acordo com 3 níveis de dificuldade (fácil com 3 pares, mediana com 6 pares e difícil com 9 pares) e com dois modos para a apresentação inicial das cartas (individualmente ou todas as cartas de uma vez). Na barra superior tem que existir a opção para alterar a quantidade de cartas apresentadas no jogo.

[RF10] – Jogo de quebra-cabeça

Será o módulo do jogo de puzzle, no qual uma ilustração é dividida por peças que serão espalhadas aleatoriamente e o utilizador tem a tarefa de arrastar as peças até a posição correta para conseguir compor a figura original. Este jogo deverá dividir uma imagem em 18 partes, fornecendo um mecanismo de ajuda para facilitar a resolução por parte dos utilizadores com certa dificuldade, por exemplo, inserir uma numeração nas peças poderá auxiliar a ordenação. Na barra superior deverá existir uma opção para trocar a figura utilizada neste jogo.

[RF11] – Jogo para completar a palavra

Será o módulo do jogo para estimular a leitura e a escrita, tem como objetivo encaixar as letras numa determinada palavra. Este jogo deverá ser composto por elementos figurativos com a funcionalidade de converter o texto para voz com a síntese de fala. Na palavra apresentada faltará uma letra e o utilizador deverá arrastar a letra para o local correto. Na barra superior existe a opção de trocar o elemento apresentado neste jogo.

Requisitos Não-Funcionais

[RNF01] – Plataforma suportada

A aplicação deverá ser desenvolvida para os dispositivos móveis (telemóvel, smartphone e tablet) existentes na plataforma Android. A versão mínima do SDK (Kit de desenvolvimento de software) será a 2.2 – Android Froyo (API nível 9). A razão da escolha deste requisito não funcional é abranger um grande número de dispositivos suportados com base na versão do sistema operativo Android.

[RNF02] – Tamanho do ficheiro da aplicação

O ficheiro binário da aplicação deverá possuir um tamanho inferior à 50 MB. Geralmente os utilizadores tendem a não instalar aplicações com um tamanho de ficheiro grande, principalmente nos mercados emergentes onde a conexão com a internet poderá ser instável.

[RNF03] – Idioma

A aplicação deverá estar disponível para a língua portuguesa.

[RNF04] – Interação com o utilizador

A aplicação deverá utilizar como controlo o ecrã tátil dos dispositivos, através de toques e gestos do utilizador. Os feedbacks obtidos pelo utilizador deverão ser através dos efeitos sonoros e a vibração do dispositivo.

[RNF05] – Suporte para diferentes dispositivos

A aplicação deverá apresentar os recursos audiovisuais (texto, imagem, áudio e vídeo) com o máximo de qualidade possível, suportando diversos modelos de dispositivos e as várias especificações existentes (processamento, memória e tamanho do ecrã).

[RNF06] – Interface gráfica e experiência do utilizador

Apresentar um esquema de cores leves que transmitam sensações como tranquilidade, serenidade, segurança e confiança. Conter elementos gráficos simples e intuitivos, não usar em excesso cores pesadas, brilhos e sombras, apresentar ícones com grande formato e bom espaçamento para facilitar a seleção e percepção do utilizador.

Proporcionar uma boa legibilidade com tipografia sem serifa, formatar o alinhamento do texto para ser justificado e definir um tamanho da fonte para uma fácil visualização.

Disponibilizar a interface gráfica com uma identidade visual, fornecer uma aparência agradável e consistente.

[RNF07] – Nível de dificuldade e sistema de recompensa e pontuações nos jogos

Os jogos não deverão apresentar nenhum mecanismo de recompensa ou pontuação e o nível de dificuldade será o mínimo possível.

O utilizador não deverá sentir nenhum sentimento negativo, como ansiedade, frustração ou desapontamento ao utilizar a aplicação.

[RNF08] – Conectividade

A aplicação não deverá utilizar conexão com a internet e não irá coletar informação do utilizador. A razão da escolha deste requisito não funcional é proteger o público infantil da internet e não recolher informações pessoais do utilizador.

[RNF09] – Remuneração, publicidade e redes sociais

A aplicação deverá ser publicada gratuitamente, sem anúncios ou publicidade e nenhuma conectividade com as redes sociais. A razão da escolha deste requisito não funcional é proteger o público infantil e evitar compras dentro da app, além das publicidades indevidas para este público e evitar o contacto com as redes sociais que é um ambiente perigoso sem a supervisão de um adulto.

[RNF10] – Distribuição

A aplicação deverá ser publicada na loja oficial da plataforma Android (Google Play Store) e em duas lojas alternativas para Android. A razão da escolha deste requisito não funcional é possibilitar a distribuição da aplicação em lojas que não possuem os bloqueios geográficos existentes na loja oficial da plataforma Android (Google Play Store) e possibilitar a instalação em dispositivos não certificados pelo Android.

4.2.2 Game Design

A elaboração do documento de game design requer um grande esforço e muitas vezes torna-se uma atividade complexa e burocrática. Para simplificar este processo foi utilizada uma metodologia denominada “Game Design Canvas”. Esta metodologia fornece um conjunto de ferramentas e processos que visam elaborar de forma clara e objetiva todas as informações necessárias para criar os requisitos de uma aplicação ou de um jogo.

Esta metodologia foi apresentada pelo autor Richard A. Carey na conferência “Serious Play Conference” em 2015, é semelhante ao modelo de negócios canvas (ferramenta de gestão estratégica), sendo dividida em 3 conjuntos principais, demonstrados na figura 4.1 [Carey, 2015]:

- **Setup:** Fornece a fundamentação do projeto, deve determinar com clareza qual é o objetivo e quem são os utilizadores.
- **Design:** Descreve quais são as principais funcionalidades, mecânicas, feedbacks, interação e experiência do utilizador.
- **Considerations:** Contém as informações sobre o produto final, como serão as fases de desenvolvimento, divulgação e distribuição.

Game Design Canvas

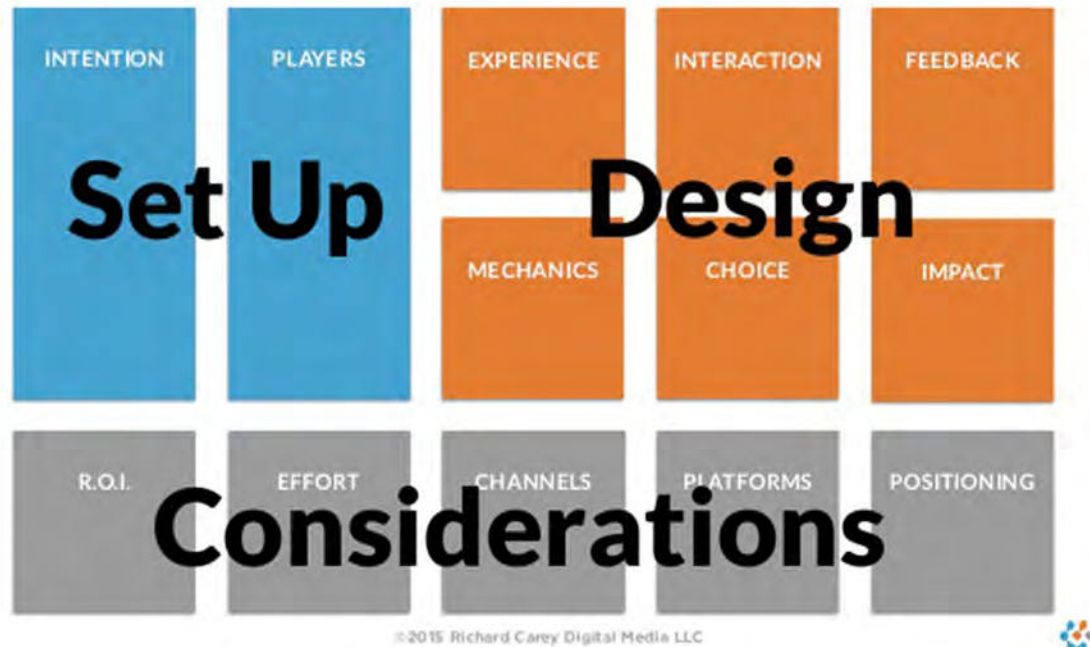


Figura 4.1: Estrutura do documento Game Design Canvas³

A figura 4.2 apresenta o “Game Design Canvas” elaborado para a aplicação:

³ **Game Design Canvas::** <https://image.slidesharecdn.com/gamedesigncanvas-seriousplay2015-150803170710-lva1-app6892/95/richard-a-carey-game-design-canvas-13-638.jpg>, acedido em 05/12/2017

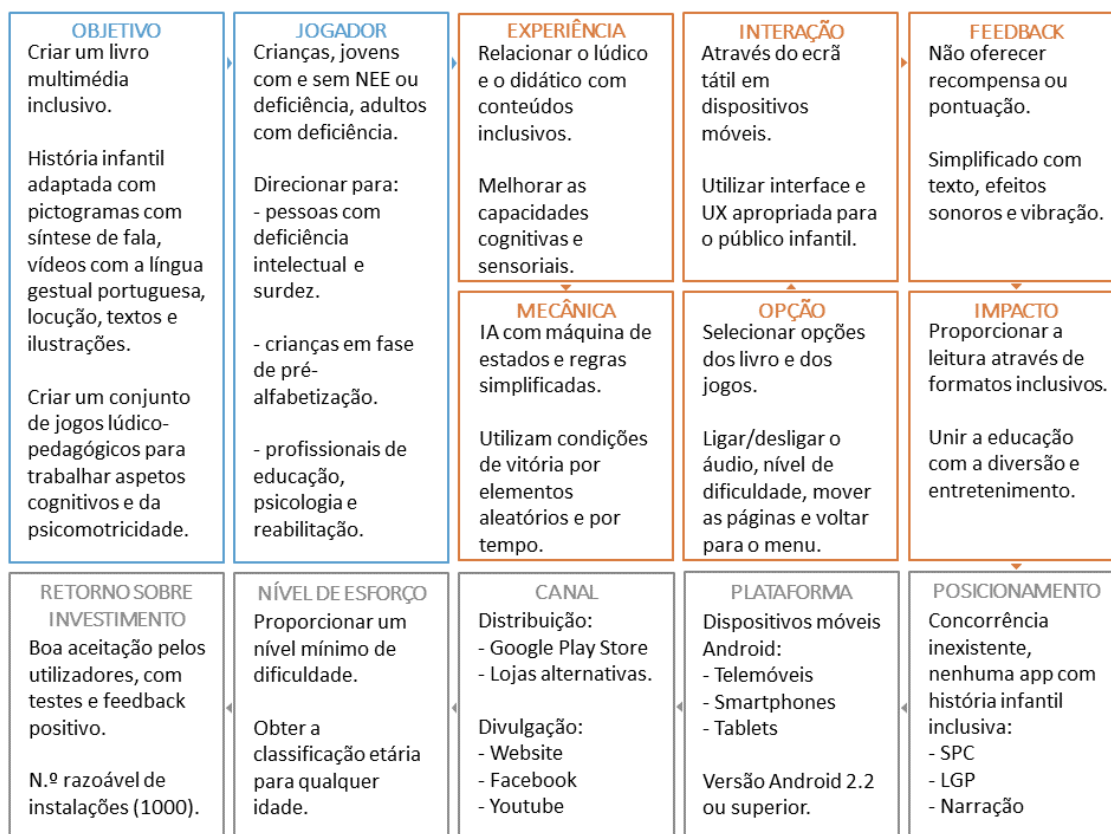


Figura 4.2: Documento Game Design Canvas da aplicação

4.2.3 Arquitetura de Software

A arquitetura selecionada durante o processo de desenvolvimento de software foi uma adaptação dos conceitos existentes na arquitetura de micro-serviços.

Este modelo é utilizado por diversas empresas como a Google, Apple, Amazon, eBay, Netflix, Uber, etc. Estas empresas chegaram à conclusão que era necessário realizar uma grande mudança no processo de desenvolvimento de software, substituindo o modelo monolítico para uma estrutura em micro-serviços [Richardson e Smith, 2016].

O modelo de arquitetura monolítico contém todas as funcionalidades agrupadas dentro de um sistema, composto por uma única unidade ou projeto. A arquitetura micro-serviços faz a divisão de um sistema em partes menores (serviços), o sistema é dividido por várias unidades ou projetos estando interligados por bibliotecas (libraries) ou API's. A figura 4.3 ilustra a diferença entre uma arquitetura de software monolítica e micro-serviços [Lewis, 2014].

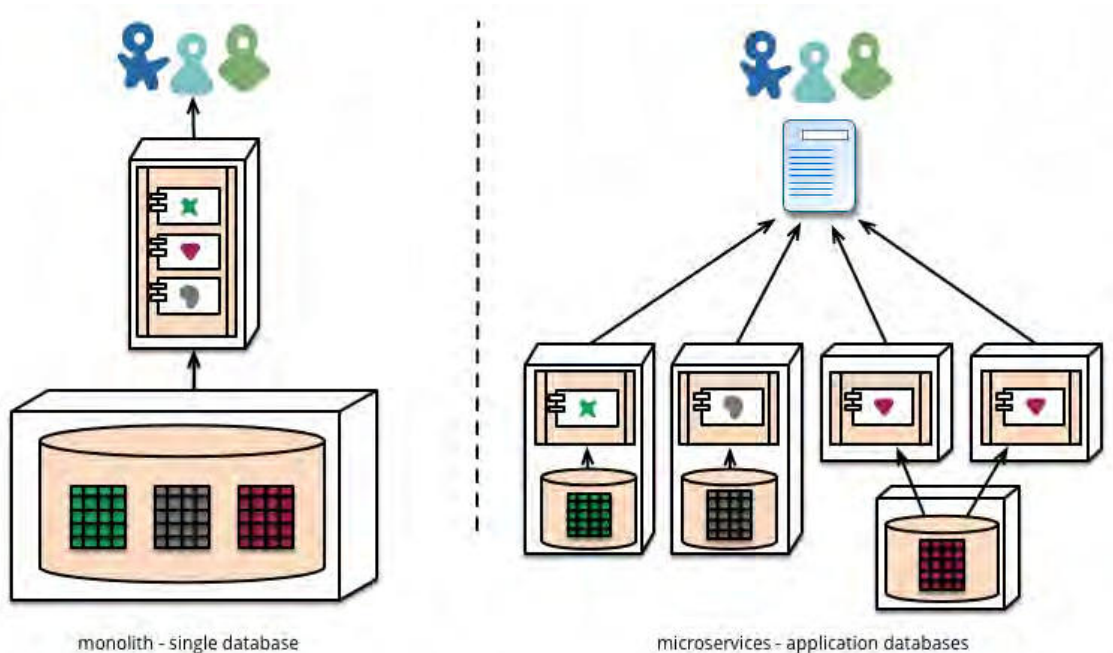


Figura 4.3: Diferença entre os modelos monolítico e micro-serviços⁴

O autor James Lewis descreve a arquitetura de micro-serviços como [Lewis, 2014]:

“The term Microservice Architecture has sprung up over the last few years to describe a particular way of designing software applications as suites of independently deployable services. While there is no precise definition of this architectural style, there are certain common characteristics around organization around business capability, automated deployment, intelligence in the endpoints, and decentralized control of languages and data.”

As principais vantagens em utilizar uma arquitetura estruturada em micro-serviços no desenvolvimento de software são [Tripoli e Carvalho, 2016]:

- Cada serviço ou módulo podem ser executados isoladamente.
- Os serviços têm apenas o tamanho necessário para isolar um determinado requisito do sistema.
- Cada serviço possui uma base de código reduzida que facilita o desenvolvimento.
- Facilidade em alterar e implantar um determinado serviço sem causar impacto no restante do sistema.

⁴ Modelos monolítico e micro-services: decentralised-data.png, acedido em 20/12/2018

<https://martinfowler.com/articles/microservices/images/decentralised-data.png>

- Cada serviço ou módulo apresenta métricas sobre si mesmo.
- O isolamento de falhas e testes são efetuados individualmente.
- O escalonamento é realizado independentemente dos demais serviços existentes no sistema.

A arquitetura por micro-serviços é uma abordagem que visa o desenvolvimento de um sistema composto por pequenas aplicações desenvolvidas, testadas e executadas isoladamente e que se interligam através da dependência de libraries ou protocolos de comunicações. Como os micro-serviços são desenvolvidos, testados e implantados de maneira independente, a escala é realizada individualmente, sem a necessidade de replicar todas as funcionalidades, conforme demonstra a figura 4.4 [Lewis, 2014]:

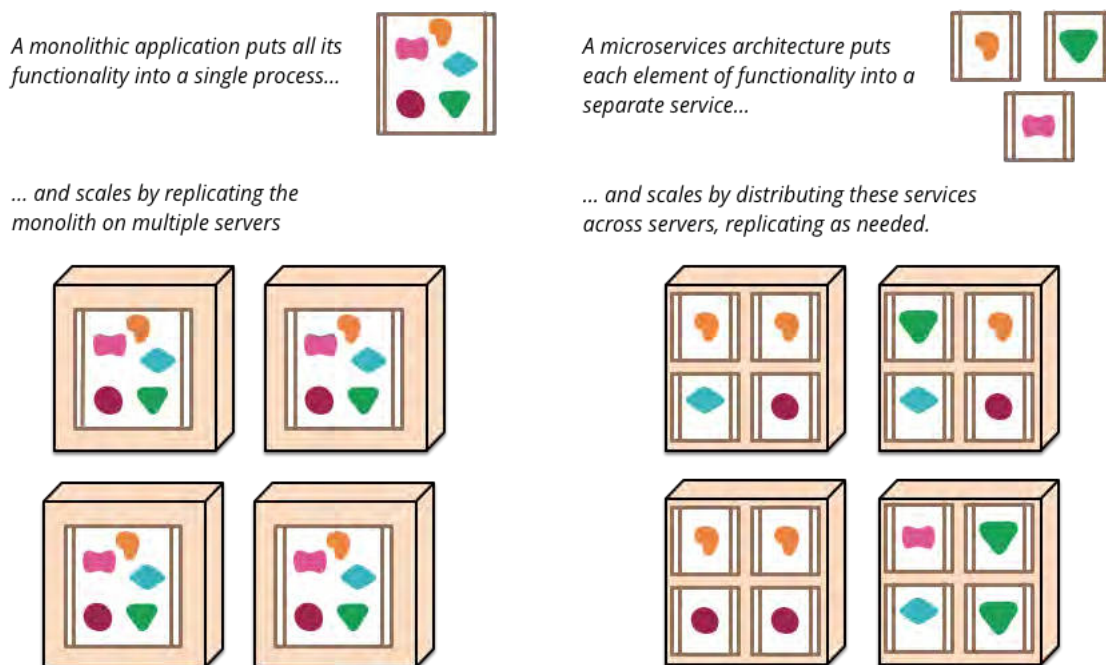


Figura 4.4: Diferença da escala entre as duas arquiteturas: monolítica e de micro serviços⁵

Apesar de ser um modelo de arquitetura antigo, um exemplo é o sistema operativo Unix que surgiu na década de 60 e utilizava este conceito, apenas nos últimos anos (a partir de 2010) este modelo passou a ser muito utilizado nas grandes empresas de tecnologia e startups, motivado principalmente pela redução da complexidade no desenvolvimento de software e facilidade em escalar o seu produto ou serviço com o menor impacto possível.

⁵ **Modelos monolítico e micro-services da escala:** <https://martinfowler.com/articles/microservices/images/sketch.png>, acedido em 25/04/2019

Muitas empresas tomaram a decisão de mudar toda a arquitetura de software do modelo monolítico para o micro-serviços, como a empresa Uber. Um caso de estudo foi realizado sobre esta mudança, disponibilizado pela autora Kappagantula⁶.

No início a decisão por desenvolver uma arquitetura monolítica fazia sentido, o serviço estava disponível apenas para uma cidade, conforme ilustra a figura 4.5.

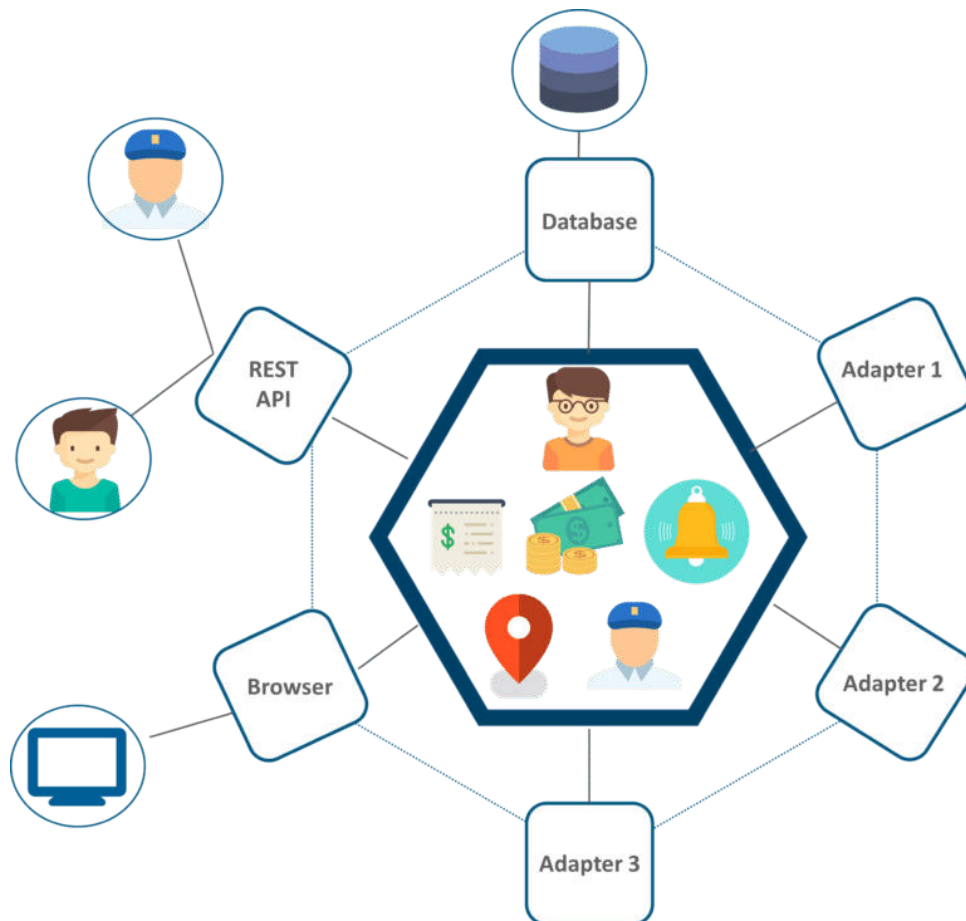


Figura 4.5: Arquitetura monolítica da Uber⁷

Os problemas apareceram quando o serviço foi expandido para todo o mundo, gerando grandes obstáculos para escalar e integrar este sistema. Os principais desafios foram:

- Todas as funcionalidades eram recompiladas, testadas e implantadas sempre que fosse introduzida uma funcionalidade nova ou se algum problema fosse corrigido.
- As correções no código-fonte ficou extremamente complicada, tudo estava num único repositório no qual diversos programadores estavam a alterar o mesmo código-fonte simultaneamente.

⁶ **Microservice Architecture – Learn, Build, and Deploy Applications:** <https://dzone.com/articles/microservice-architecture-learn-build-and-deploy-a>, acessado em 25/04/2019

⁷ **Arquitetura monolítica da Uber:** <https://d1jnx9ba8s6j9r.cloudfront.net/blog/wp-content/uploads/2018/02/Monolithic-Architecture-Of-UBER-Microservice-Architecture-Edureka-768x730.png>, acessado em 20/12/2018

A solução desenvolvida pela empresa foi dividir o sistema monolítico existente em vários micros-serviços, criando vários repositórios de código-fonte de acordo com as funcionalidades ou serviços existentes, conforme demonstra a figura 4.6. As principais vantagens foram:

- Implementação de unidades isoladas dos serviços existentes na aplicação, sendo desenvolvidas, executadas e implantadas individualmente, eliminando as várias ocorrências de conflito no código-fonte.
- Todas as funcionalidades podem ser escaladas individualmente, não existindo interdependência.

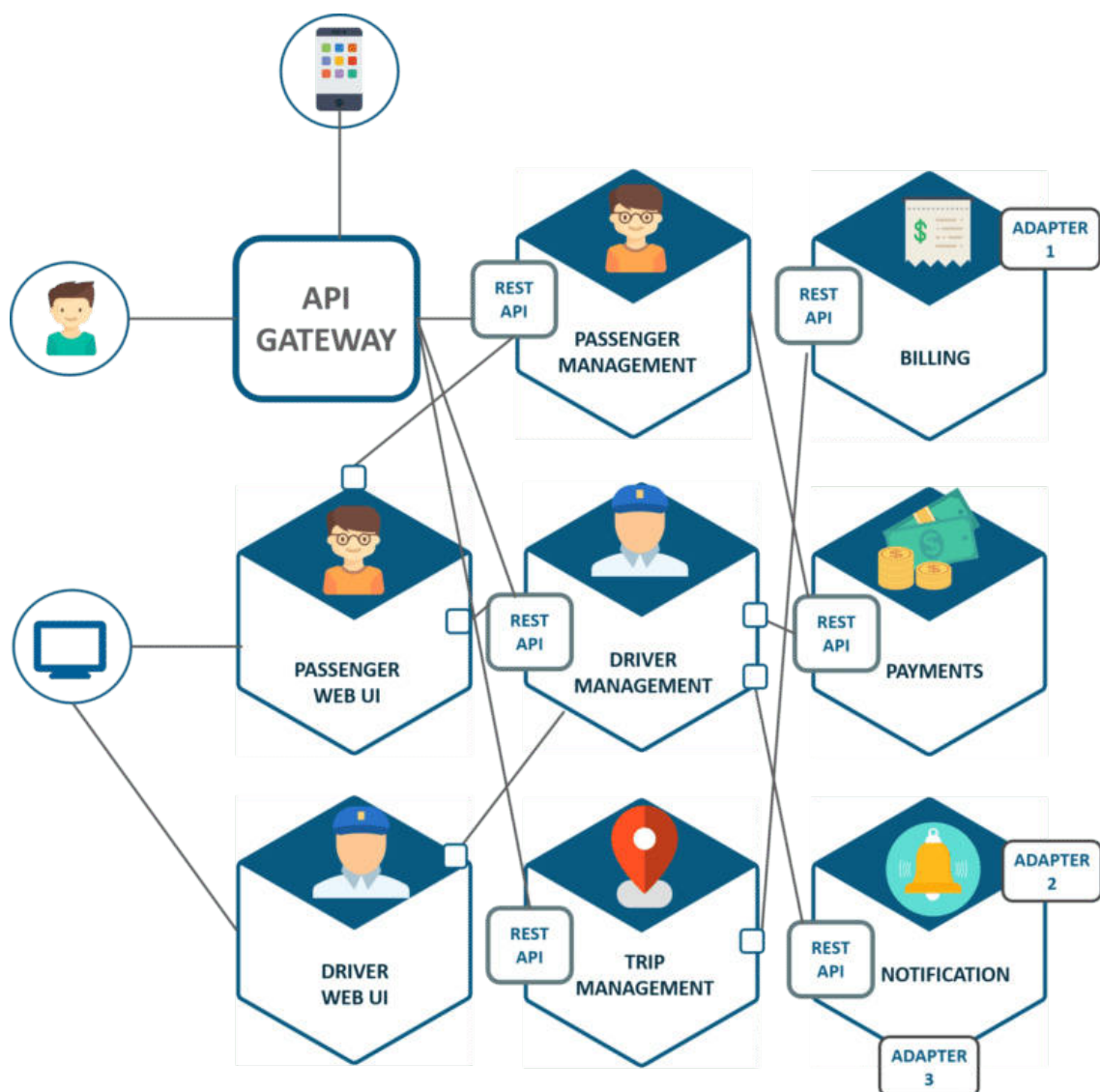


Figura 4.6: Arquitetura micro-serviços da Uber⁸

⁸ **Arquitetura micro-serviços da Uber:** <https://d1jnx9ba8s6j9r.cloudfront.net/blog/wp-content/uploads/2018/02/Microservice-Architecture-Of-UBER-Microservice-Architecture-Edureka-768x762.png>, acessado em 25/04/2019

Para exemplificar os benefícios no caso da Uber, supondo que seja necessária realizar uma alteração no sistema de pagamento, na arquitetura atual basta realizar a alteração isoladamente neste módulo e implantá-lo sem impactar os outros serviços, na arquitetura monolítica seria necessário alterar todo o sistema.

Uma boa prática no desenvolvimento avançado para a plataforma Android é dividir uma aplicação em componentes funcionais. Por exemplo, ao desenvolver uma aplicação com um sistema de autenticação do utilizador com informações localizadas numa base de dados externa. Ao utilizar a arquitetura por micro-serviços, a aplicação seria dividida em dois módulos, um sendo a autenticação e o outro com a aplicação principal. O processo de desenvolvimento seria menos complexo com funcionalidades isoladas e com o código-fonte reduzido, possibilitando uma redução no esforço durante o processo de compilação e implantação. Se no futuro houver a necessidade de uma mudança na base de dados, será exigido apenas a alteração no módulo de autenticação, não será alterado a aplicação principal, apenas a dependência do módulo de autenticação será atualizada.

4.2.4 Plataforma Android

A plataforma selecionada para o desenvolvimento deste projeto é a Google Android. A escolha deve-se à vários motivos, disponibilizadas pela Google através do website “O Android é para todos”⁹. A seguir são demonstradas as principais razões:

- É uma plataforma gratuita e possui o código aberto, fornece uma ampla quantidade de documentação, materiais de estudo e ferramentas para auxiliar o processo de desenvolvimento de software.
- Fornece todos os recursos necessários para o desenvolvimento com um completo conjunto de software (SDK, IDE, simuladores e API's).
- Tem uma enorme variedade de dispositivos (mais de 24.000 modelos diferentes já produzidos) e diversas empresas que fabricam equipamentos com este sistema operativo (mais de 1.300 marcas diferentes), fornecendo uma maior diversidade nas especificações, dando a oportunidade de comercializá-los com preços acessíveis.
- Facilidade no processo de publicação e distribuição através da loja de aplicações Google Play Store.

⁹ O Android é para todos : https://www.android.com/intl/pt_pt/everyone/, acedido em 20/12/2018

No desenvolvimento de software convencional, geralmente existe um único ponto de entrada executados como um processo monolítico. Quando se desenvolve aplicações para dispositivos móveis a estrutura é mais complexa, por exemplo uma aplicação pode conter vários componentes, incluindo atividades, fragmentos, serviços, provedores de conteúdo e broadcast receivers.

Neste projeto foi realizada uma analogia com o desenvolvimento de uma aplicação modular. Foram definidas 3 camadas principais, sendo os módulos Framework na primeira camada, Recursos na segunda e a terceira é a Aplicação final composta pelos micro-serviços.

4.3 Implementação da Aplicação

A aplicação é composta por um conjunto de micro-aplicações ou módulos (micro-serviços), cada um sendo implementado e testado isoladamente, e no final são adaptados em formato de biblioteca e integrado numa única aplicação.

O processo de compilação foi automatizado para que com o mínimo de alterações no projeto uma micro-aplicação isolada que se torna uma biblioteca (*library*) sendo acoplada na aplicação final.

A seguir a figura 4.7 demonstra como foi elaborada a estrutura da arquitetura durante o desenvolvimento desta aplicação. As ligações demonstram a regra de dependência, o módulo Recursos possui a dependência do Framework e os micro-serviços possuem a dependência do módulo Recursos que herda a dependência do Framework e a aplicação final importa os micro-serviços.

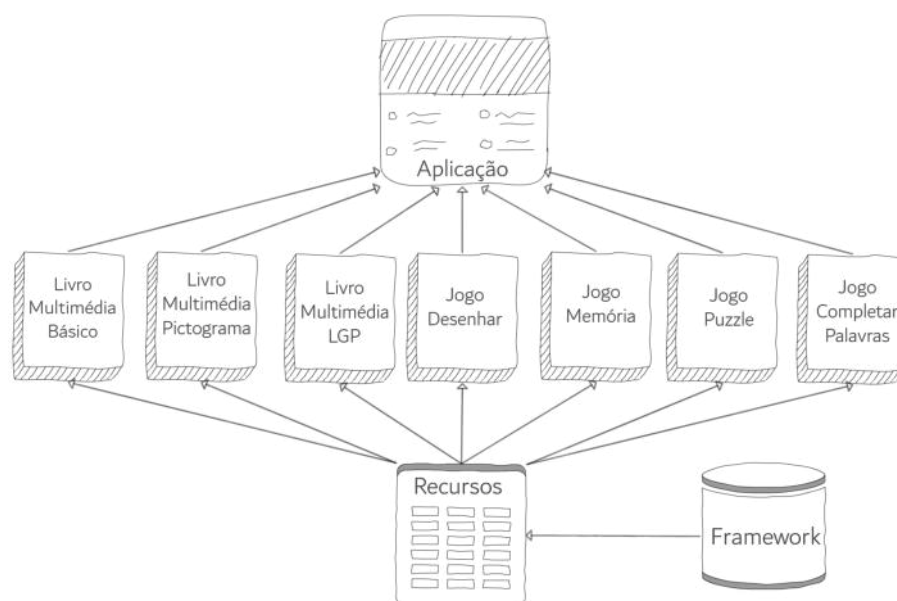


Figura 4.7: Arquitetura desenvolvida para a aplicação

Ao nível da programação para a plataforma Android a arquitetura foi implementada ao criar um projeto chamado Framework, sendo configurado para ser um módulo do tipo library e não uma aplicação. Posteriormente foram desenvolvidos o projeto Recursos e os módulos híbridos (micro-aplicações) que funcionam como aplicações isoladas e como libraries.

Para se criar um módulo que seja uma library é necessário fazer ajustes no ficheiro de configuração do projeto no Android Studio e criar um repositório onde serão alojados todos os módulos desenvolvidos.

No desenvolvimento para a plataforma Android é possível ter 3 tipos de dependência modular:

- **Dependência de módulo local:** Declara uma dependência de um módulo do Android chamado dentro do aplicativo, no qual o sistema compila o módulo e empacota o conteúdo compilado resultante na aplicação final (ficheiro APK).
- **Dependência binária local:** Declara dependências em ficheiros do tipo JAR dentro do diretório do projeto da aplicação.
- **Dependência binária remota:** Declara dependências de módulos de acordo com a versão da library e dentro de um namespace específico.

O código a seguir demonstra como é a configuração de uma aplicação que utilize os 3 tipos de dependência modular localizado no ficheiro de compilação build.gradle:

```
1  apply plugin: 'com.android.application'
2
3  android { ... }
4
5  dependencies {
6
7      // Dependência de módulo local
8      implementation project(":mylib")
9
10     // Dependência binária local
11     implementation fileTree(dir: 'libs', include: ['*.jar'])
12
13     // Dependência binária remota
14     implementation 'com.example.android:mylib:1.0.0'
15 }
```

Neste projeto foi criado um repositório com a dependência remota, é uma boa prática no desenvolvimento, pois facilita o reuso e reduz a complexidade no código-fonte ao isolar remotamente os módulos.

A seguir será demonstrado como um exemplo os passos para criar e configurar o módulo Framework:

1. **Criar o projeto no Android Studio:** Criar um projeto conforme demonstram as figuras 4.8, 4.9, 4.10 e 4.11:

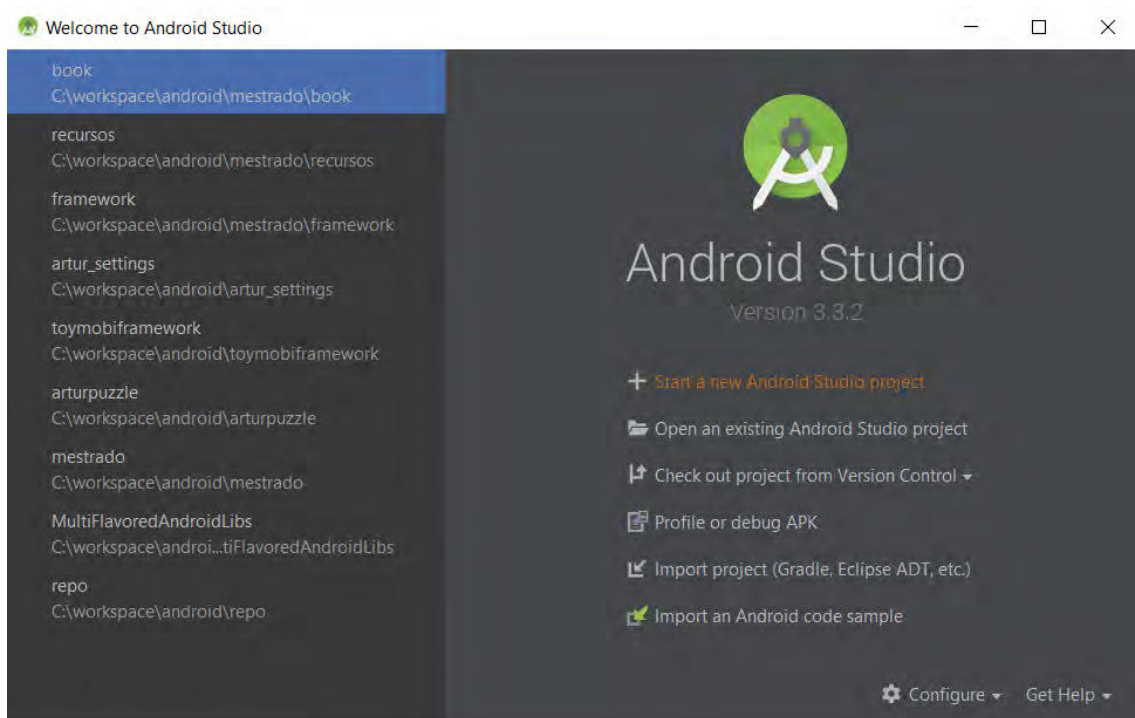


Figura 4.8: Criar projeto no Android Studio, passo inicial

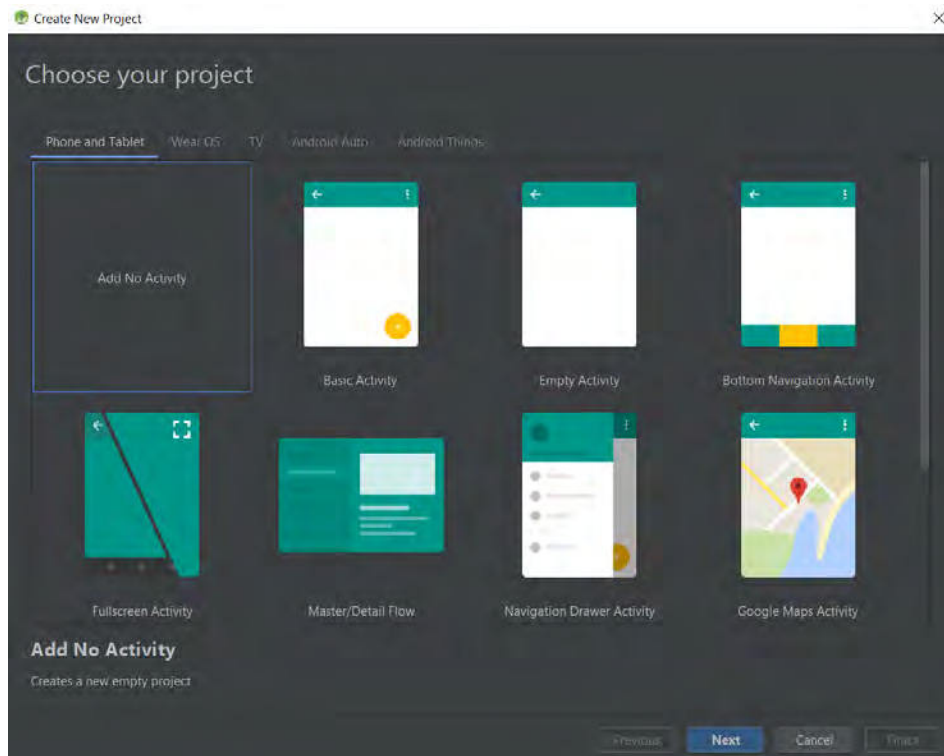


Figura 4.9: Criar projeto no Android Studio, selecionar o tipo de projeto

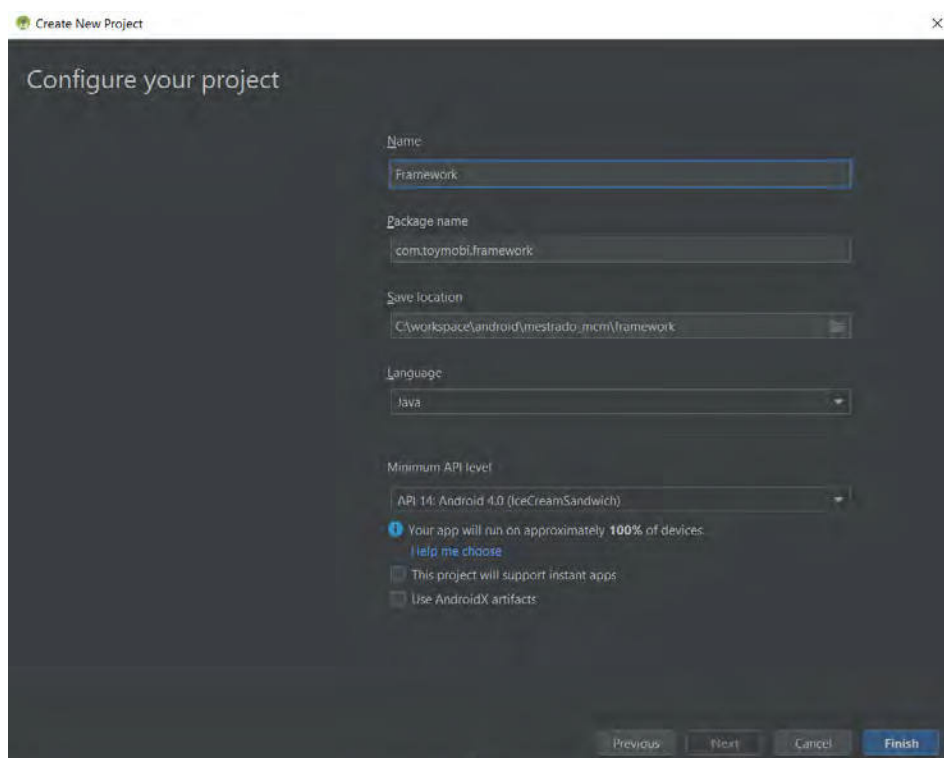


Figura 4.10: Criar projeto no Android Studio, configuração inicial do projeto

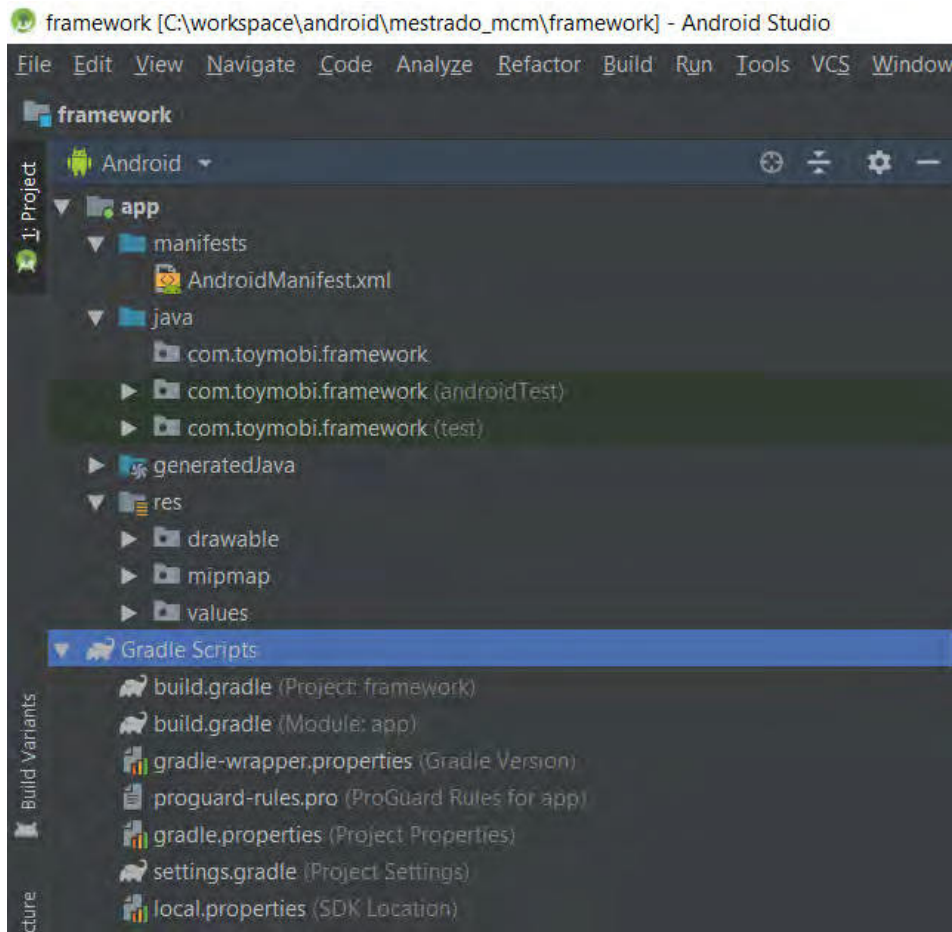


Figura 4.11: Criar projeto no Android Studio, estrutura do projeto

2. **Adicionar as propriedades:** Deve ser adicionada as propriedades demonstradas a seguir dentro do ficheiro `gradle.properties`, são utilizadas para criar o namespace do módulo:

```

1 POM_ARTIFACT_ID=framework
2 VERSION_NAME=1.0.0
3 GROUP=com.toymobi

```

3. **Editar as configurações de compilação:** Deve ser editado o ficheiro de compilação do projeto, o `build.gradle`. O primeiro ajuste é para transformar o projeto numa library, substituindo `apply plugin: 'com.android.application'` para `apply plugin: 'com.android.library'` e remover o `applicationId "com.toymobi.framework"`, além de inserir o plugin Maven, adicionando a linha `apply plugin: 'maven'`, conforme demonstra o código a seguir:

```

1 apply plugin: 'com.android.library'
2 apply plugin: 'maven'
3
4 android {
5     compileSdkVersion 28
6
7     defaultConfig {

```



```

8         //applicationId "com.toymobi.framework"
9         minSdkVersion 14
10        targetSdkVersion 28
11        versionCode 1
12        versionName "1.0"
13        vectorDrawables.useSupportLibrary = true
14    }

```

4. **Inserir repositório nas configurações de compilação:** No ficheiro de compilação do projeto é necessário criar uma task chamada `uploadArchives` utilizada para copiar o módulo para o repositório criado, será uma pasta local chamada `repo` com as propriedades de configuração do módulo, conforme demonstra o trecho do código a seguir:

```

1  uploadArchives {
2
3      repositories {
4          mavenDeployer {
5              repository(url: "file:///C:/workspace/android/mestrado/repo")
6              pom.groupId = GROUP
7              pom.artifactId = POM_ARTIFACT_ID
8              pom.version = VERSION_NAME
9          }
10     }
11 }
12 task install(dependsOn: uploadArchives)

```

A seguir está o código completo com as alterações no ficheiro de configuração do `build.gradle`:

```

1  apply plugin: 'com.android.library'
2  apply plugin: 'maven'
3
4  android {
5      compileSdkVersion 28
6
7      defaultConfig {
8          minSdkVersion 14
9          targetSdkVersion 28
10         versionCode 1
11         versionName "1.0"
12         vectorDrawables.useSupportLibrary = true
13     }
14
15     buildTypes {
16
17         release {
18             shrinkResources false
19             minifyEnabled true
20             zipAlignEnabled true
21             debuggable false
22             jniDebuggable false
23             proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), '
                proguard-rules.pro', 'proguard-rules-disable-logging.pro'
14         }

```

```

25
26     debug {
27         debuggable true
28         jniDebuggable true
29     }
30 }
31
32     dexOptions {
33         javaMaxHeapSize "4g"
34     }
35 }
36
37 uploadArchives {
38
39     repositories {
40         mavenDeployer {
41             repository(url: "file:///C:/workspace/android/mestrado/repo")
42             pom.groupId = GROUP
43             pom.artifactId = POM_ARTIFACT_ID
44             pom.version = VERSION_NAME
45         }
46     }
47 }
48 task install(dependsOn: uploadArchives)
49
50 dependencies {
51     implementation fileTree(dir: 'libs', include: ['*.jar'])
52     implementation 'androidx.appcompat:appcompat:1.1.0-alpha04'
53     implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
54 }

```

5. **Compilar e gerar o módulo:** Após compilar o projeto, basta executar a task install através da opção Run Configuration no gradle, como demonstra a figura 4.12 ou através da linha de comando gradlew install.

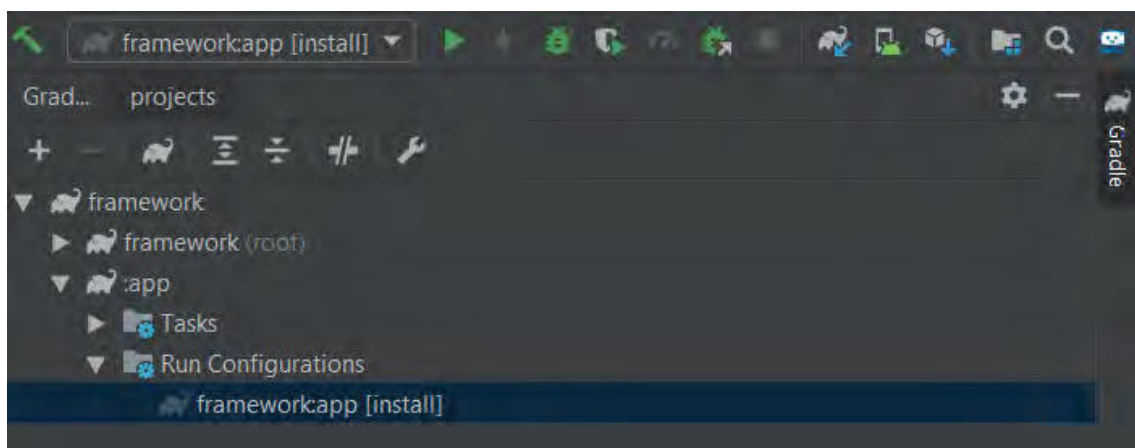


Figura 4.12: Opção para executar a task install no gradle

No processo de implantação é criada uma pasta com a library gerada pelo projeto Framework dentro do repositório. O módulo é empacotado num ficheiro com o formato Android Archive (AAR) que pode ser utilizado como dependência de outro módulo ou aplicação para a plataforma Android. Os ficheiros AAR podem conter recursos da plataforma Android como layouts, imagens, textos, dimensões, temas, cores, etc. além de classes e métodos Java.

O módulo Recursos foi criado da mesma forma do Framework, a única diferença está na dependência do módulo Framework e caminho do repositório, conforme o trecho de código a seguir:

```
1  android { ... }
2
3  repositories {
4      maven {
5          url "file:///C:/workspace/android/mestrado/repo/"
6      }
7  }
8
9  dependencies {
10     implementation fileTree(dir: 'libs', include: ['*.jar'])
11     implementation 'com.toymobi:framework:1.0.0'
12 }
```

Para criar um projeto Android híbrido (funciona como uma aplicação isolada e como uma library) é através da criação de um módulo interno, deve-se ter cuidado na forma de estruturar o projeto para que ambos funcionem. A library tem que estar implementada para facilitar o acesso por outros módulos, neste projeto foi estruturada para carregar um fragmento em outros projetos.

Os passos para se criar o projeto no Android Studio é o mesmo citado anteriormente. A diferença será a inclusão de um módulo dentro do projeto para executar a aplicação, conforme demonstrado a seguir:

1. **Criar o projeto no Android Studio:** É conforme demonstrado anteriormente, sendo necessário alterar as configurações no ficheiro de compilação para transformar o projeto numa library, adicionar as propriedades no ficheiro gradle.properties, configurar o repositório. Após realizar os passos anteriores, a estrutura do projeto será como demonstra a figura 4.13.

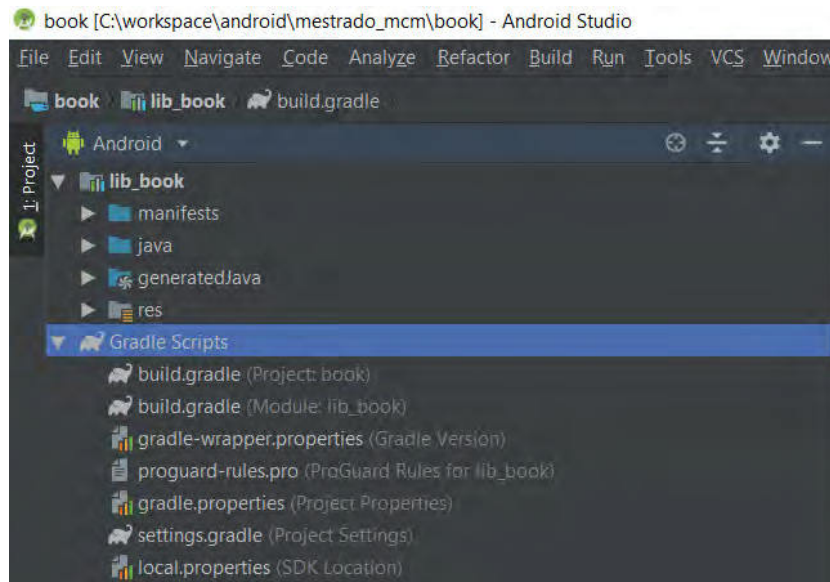


Figura 4.13: Criar projeto híbrido no Android Studio, passo inicial criando a library

2. **Criar o um novo módulo dentro do projeto:** O próximo passo será a criação de um módulo do tipo Phone & Tablet no Android Studio. Basta seleccionar a opção File->New->New Module, conforme demonstra a figura 4.14:

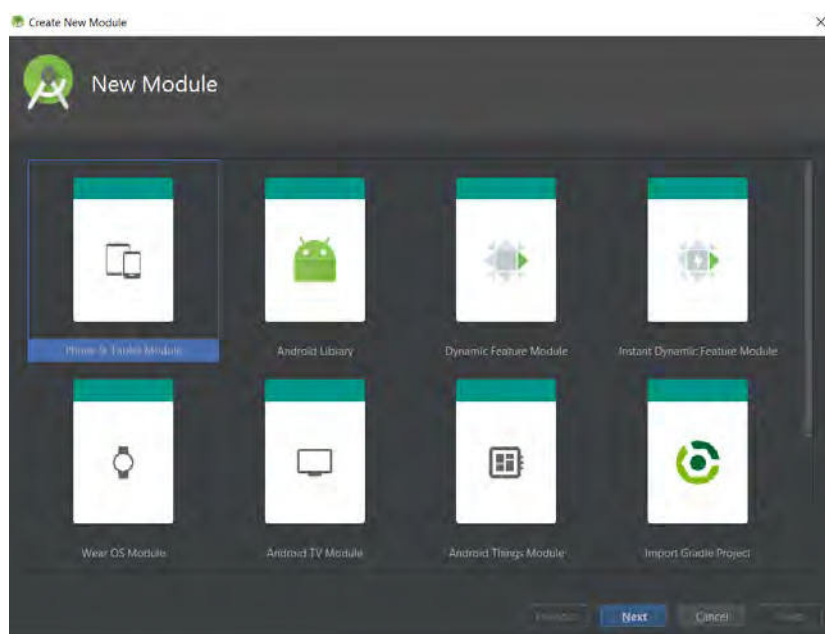


Figura 4.14: Criar módulo dentro do projeto no Android Studio

3. **Alteração na estrutura do projeto:** Após criar o módulo é necessário modificar a estrutura de compilação do projeto, é o mesmo passo citado anteriormente, conforme demonstra o código a seguir.

```
1 // build.gradle do módulo library
2 apply plugin: 'com.android.library'
3 apply plugin: 'maven'
4
5 android {
6     compileSdkVersion 28
7     defaultConfig {
8         minSdkVersion 14
9         targetSdkVersion 28
10        versionCode 1
11        versionName "1.0"
12    }
13    { ... }
14    dependencies {
15        implementation fileTree(dir: 'libs', include: ['*.jar'])
16        implementation 'com.toymobi:recursos:1.0.0'
17    }
18
19 // build.gradle do módulo app
20
21 apply plugin: 'com.android.application'
22 apply plugin: 'maven'
23
24 android {
25     compileSdkVersion 28
26     defaultConfig {
27         applicationId "com.toymobi.book_app"
28         minSdkVersion 14
29         targetSdkVersion 28
30         versionCode 1
31         versionName "1.0"
32    }
33    { ... }
34
35    dependencies {
36        implementation fileTree(dir: 'libs', include: ['*.jar'])
37        implementation project(':book_lib')
38        implementation 'com.toymobi:recursos:1.0.0'
39    }
40 }
```

A figura 4.15 demonstra como será a estrutura do módulo com uma library e uma aplicação.

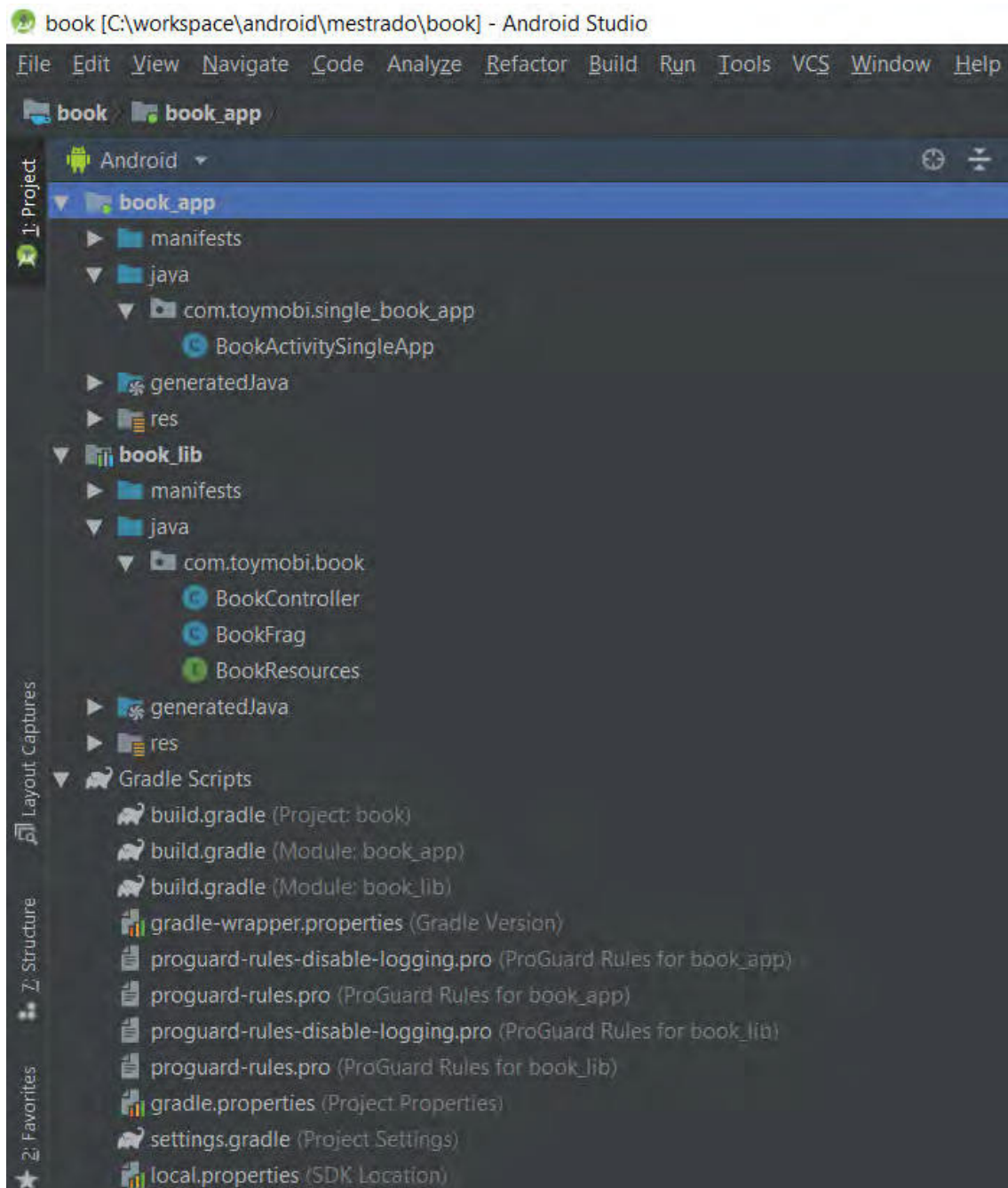


Figura 4.15: Estrutura do projeto no Android Studio com a library e a aplicação

A library não possui nenhuma Activity e a classe principal que será utilizada pela aplicação é o Fragment que será chamado dentro da Activity na aplicação para executar a aplicação.

```
1 // manifest do módulo app
2 <?xml version="1.0" encoding="utf-8"?>
3 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
4 xmlns:tools="http://schemas.android.com/tools"
5 package="com.toymobi.single_book_app">
6
7 <uses-permission android:name="android.permission.VIBRATE" />
8
9 <application
10     android:allowBackup="true"
11     android:icon="@drawable/icon_app"
12     android:label="@string/book_name"
13     android:largeHeap="true"
14     android:logo="@drawable/icon_app"
15     android:screenOrientation="landscape"
16     android:theme="@style/Theme.NoBackground"
17     tools:ignore="GoogleAppIndexingWarning">
18
19     <activity
20         android:name=".BookActivitySingleApp"
21         android:configChanges="locale|orientation|keyboardHidden"
22         android:label="@string/book_name"
23         android:screenOrientation="landscape"
24         android:theme="@style/CustomActionBarTheme">
25         <intent-filter>
26             <action android:name="android.intent.action.MAIN" />
27
28             <category android:name="android.intent.category.LAUNCHER" />
29         </intent-filter>
30     </activity>
31
32 </application>
33
34 </manifest>

1 public class BookActivitySingleApp extends AppCompatActivity {
2
3     @Override
4     protected void onCreate(Bundle savedInstanceState) {
5         super.onCreate(savedInstanceState);
6         setContentView(com.toymobi.book.R.layout.main_layout_book);
7         startBookFrag();
8     }
9
10    private final void startBookFrag() {
11
12        final FragmentManager fragmentManager = getSupportFragmentManager();
13
14        final FragmentTransaction fragmentTransaction = fragmentManager.
15            beginTransaction();
16
17        BookFrag fragmentBook = (BookFrag) fragmentManager.findFragmentByTag(
18            BookFrag.FRAGMENT_TAG_BOOK_TEXT);
19
20        if (fragmentBook == null) {
```



```

19         fragmentBook = new BookFrag();
20         fragmentTransaction.add(com.toymobi.book.R.id.main_layout_fragment_id,
                                fragmentBook, BookFrag.FRAGMENT_TAG_BOOK_TEXT);
21         fragmentTransaction.commit();
22     }
23 }
24 }

```

4. **Compilar e executar:** Nas opções do gradle basta selecionar o módulo para compilar e executar a aplicação, conforme demonstra a figura 4.16.

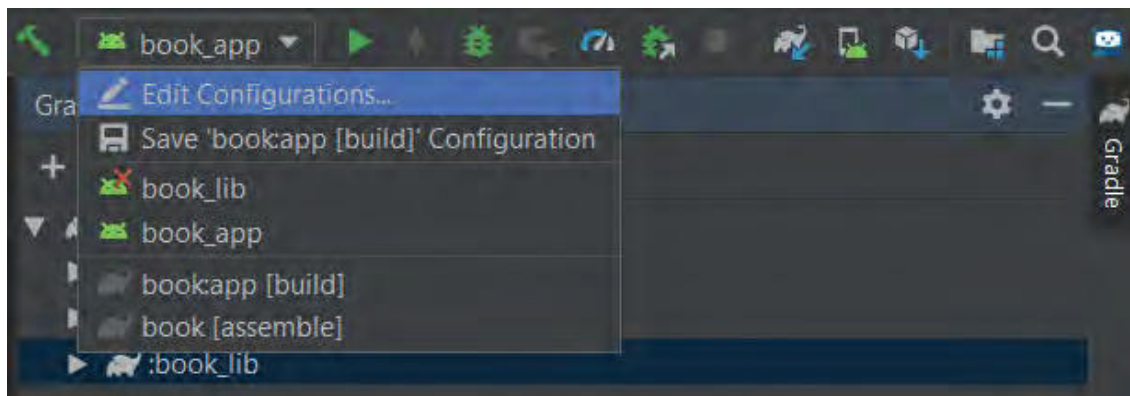


Figura 4.16: Opções do gradle para compilar e executar projeto com a aplicação e a library

A arquitetura interna de cada módulo não foi desenvolvida com base numa metodologia específica, como por exemplo a MVC (Model View Control). Procurou-se desenvolver uma arquitetura por separação de conceitos, por exemplo, na redução do tamanho e complexidade das classes Activity e Fragment. Geralmente os módulos desenvolvidos neste projeto segue um padrão de desenvolvimento com 3 classes principais:

- **Activity**¹⁰: Esta classe fornece meios para os utilizadores interagirem com as aplicações, são modulares e possuem um ciclo de vida definido. Fornecem para o utilizador uma interface agrupadas, atingindo as funcionalidades e tarefas pretendidas por uma aplicação.
- **Fragment**¹¹: Um objeto da classe Fragment representa o comportamento de uma parte da interface do utilizador, sendo chamado dentro de uma classe Activity. É possível ter vários componentes do tipo Fragment na mesma Activity. É considerada como uma entidade modular de uma Activity, possuindo o seu próprio ciclo de vida, entrada e saída, sendo adicionada ou removida sem dificuldade.

¹⁰ **Classe Activity**:: <https://developer.android.com/reference/android/app/Activity.html>, acedido em 25/04/2019

¹¹ **Classe Fragment**:: <https://developer.android.com/reference/android/app/Fragment.html>, acedido em 25/04/2019

- **Controller:** Esta classe foi criada para conter toda a lógica funcional de acordo com cada módulo da aplicação. Foi implementada para ser uma parte importante dos requisitos funcionais de cada módulo, deixando as classes Activity e Fragment apenas com as funcionalidades essenciais para o funcionamento das mesmas.

A seguir estão os exemplos de código-fonte das classes Activity, Fragment e Controller desenvolvidos no módulo de leitura. Este micro-serviço se comporta como uma library e pode ser executado isoladamente como uma aplicação.

A classe Activity deste módulo chama-se BookActivitySingleApp, sobrescrevendo a implementação dos métodos do ciclo de vida do fragmento, sendo o método mais importante o onCreateView, onde é criada a instância do Controller, conforme demonstra o exemplo de código-fonte a seguir:

```

1 public class BookActivitySingleApp extends AppCompatActivity {
2     @Override
3     protected void onCreate(Bundle savedInstanceState) {
4         super.onCreate(savedInstanceState);
5         setContentView(com.toymobi.book.R.layout.main_layout_book);
6         startBookFrag();
7     }
8
9     private final void startBookFrag() { ... }
10 }

```

A classe Fragment deste módulo chama-se BookFrag, sobrescrevendo a implementação dos métodos do ciclo de vida do fragmento, sendo o método mais importante o onCreateView, onde é criado o objeto Controller, conforme demonstra o exemplo de código-fonte a seguir:

```

1 public class BookFrag extends Fragment {
2
3     private SimpleSoundPool sfx;
4
5     private MusicBackgroundController musicBackgroundController;
6
7     private BookController bookController;
8
9     private VibrationFeedback vibrationFeedback;
10
11     @Override
12     public final void onCreate(final Bundle savedInstanceState) { ... }
13
14     @Override
15     public void onResume() { ... }
16
17     @Override
18     public void onPause() { ... }
19
20     @Override
21     public final void onStop() { ... }
22 }

```

```

23     @Override
24     public void onDestroy() { ... }
25
26     @Override
27     public final void onCreateOptionsMenu(final Menu menu,
28                                         final MenuInflater inflater) { ... }
29
30     @Override
31     public final boolean onOptionsItemSelected(final MenuItem item) { ... }
32
33     @Override
34     public final View onCreateView(final LayoutInflater inflater,
35                                   final ViewGroup container,
36                                   final Bundle savedInstanceState) {
37
38         final View view = inflater.inflate(R.layout.layout_viewpager, container, false);
39
40         if (bookController == null) {
41             bookController = new BookController(getActivity());
42         }
43         ( ... )
44     }
45
46     private final void createActionBar() { ... }
47
48     private void createSFX() { ... }
49
50     private void deallocate() { ... }
51
52     public final void playFeedbackButton() { ... }
53 }

```

4.3.1 Aplicação – O menino que tinha medo do escuro

A aplicação contém o acoplamento dos módulos desenvolvidos como micro-serviços isolados e acessível por todas as libraries.

O ecrã inicial (*splash screen*) apresenta duas transições até chegar ao menu principal, onde estão localizados todos os itens de seleção dos módulos, além das opções para encerrar a aplicação, informação, ativar e desativar o áudio.

A interface principal é composta por um componente (*container*) para encaixar o respetivo fragmento de cada módulo selecionado pelo utilizador, como demonstram as figuras 4.17 e 4.18.



Figura 4.17: Ecrã inicial e menu principal I



Figura 4.18: Ecrã inicial e menu principal II

Este módulo é a aplicação final corresponde com o **RF01 – Seleção dos módulos**, utiliza a dependência dos outros micro-serviços. Possui layouts simplificados, sendo um utilizado para o ecrã inicial com a ilustração da capa e o nome do livro, e outro layout contém uma grelha com as opções de seleção de cada requisito funcional que compõe a aplicação.

A seguir está como exemplo um código-fonte da classe Fragment onde estão implementadas as principais funcionalidades deste módulo.

O método `createStart` é responsável por criar um evento de clique (`OnClickListener`) para as 7 opções existentes no menu principal, que ao clicar é criado um fragmento de acordo com o módulo selecionado, conforme demonstra o exemplo de código-fonte a seguir:

```
1 public class MainMenuFragment extends Fragment {
2
3     private SimpleSoundPool sfx;
4
5     private VibrationFeedback vibrationFeedback;
6
7     private MusicBackgroundController musicBackgroundController;
8
9     private static final int[] options = { R.id.option_1, R.id.option_2,
10         R.id.option_3, R.id.option_4, R.id.option_5, R.id.option_6,
11         R.id.option_7 };
12
13     public static final String MAIN_MENU_FRAGMENT_TAG = "MAIN_MENU_FRAGMENT";
14
15     private OnClickListener optionOnClickListener;
16
17     private CharSequence pageInfoText = null;
18
19     @Override
20     public final View onCreateView (final LayoutInflater inflater,
21         final ViewGroup container, final Bundle savedInstanceState) { ... }
22
23     @Override
24     public void onCreate (Bundle savedInstanceState) { ... }
25
26     @Override
27     public final void onCreateOptionsMenu (final Menu menu,
28         final MenuInflater inflater) { ... }
29
30     @Override
31     public boolean onOptionsItemSelected (final MenuItem item) { ... }
32
33     private void goToInfo () { ... }
34
35     @Override
36     public final void onResume () { ... }
37
38     @Override
39     public final void onPause () { ... }
40
41     @Override
42     public final void onStop () { ... }
43 }
```

```

44     @Override
45     public final void onDestroy () { ... }
46
47     private void createActionBar () { ... }
48
49     public final void playFeedbackButton () { ... }
50
51     private void createStart (final View view) {
52
53         ( ... )
54
55         if (optionOnClickListener == null) {
56
57             optionOnClickListener = new OnClickListener() {
58
59                 @Override
60                 public void onClick (final View view) {
61                     if (view != null) {
62                         final int viewID = view.getId();
63
64                         switch (viewID) {
65                             case R.id.option_1:
66                                 goOption1();
67                                 break;
68                             case R.id.option_2:
69                                 goOption2();
70                                 break;
71                             case R.id.option_3:
72                                 goOption3();
73                                 break;
74                             case R.id.option_4:
75                                 goOption4();
76                                 break;
77                             case R.id.option_5:
78                                 goToOption5();
79                                 break;
80                             case R.id.option_6:
81                                 goToOption6();
82                                 break;
83                             case R.id.option_7:
84                                 goToOption7();
85                                 break;
86                         }
87                     }
88                 }
89             };
90         }
91
92         for (int i = 0; i < options.length; i++) {
93             final View optionView = view.findViewById(options[i]);
94
95             if (optionView != null && optionOnClickListener != null) {
96                 optionView.setOnClickListener(optionOnClickListener);
97             }
98         }
99     }
100
101     private void goOption1 () { ... }
102

```

```

103     private void goOption2 () { ... }
104
105     private void goOption3 () { ... }
106
107     private void goOption4 () { ... }
108
109     private void goOption5 () { ... }
110
111     private void goOption6 () { ... }
112
113     private void deallocate () { ... }
114
115 }

```

4.3.2 Módulo – Livro Multimédia Básico

Este módulo contém o livro multimédia com texto, áudio e imagem. A interface está organizada com uma caixa de texto localizada na parte esquerda com a opção de *scroll* na horizontal quando ultrapassar o limite disponível para o texto, no lado direito está uma imagem com a ilustração. A transição entre das páginas faz-se com o movimento de deslizar do dedo no ecrã e através dos botões para avançar e retroceder localizados na barra superior da aplicação, como demonstram as figuras 4.19 e 4.20.



Figura 4.19: Livro multimédia I

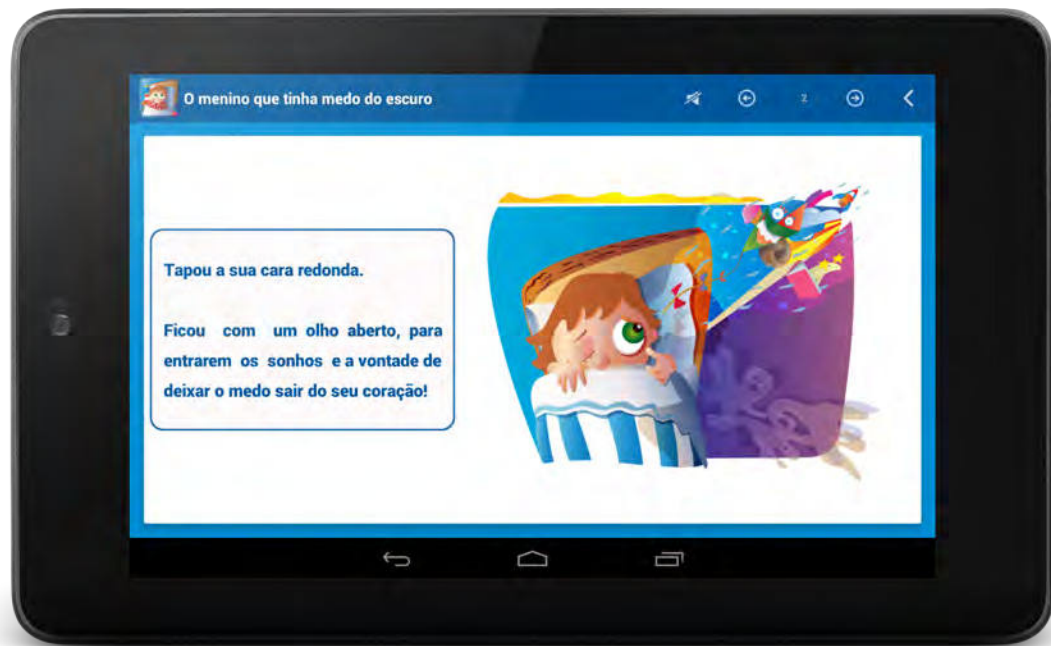


Figura 4.20: Livro multimédia II

Este módulo híbrido funciona como library e aplicação, corresponde com o **RF05 – Livro multimédia básico**, possui um conjunto de layouts para cada página da história e que reutiliza 2 layouts, sendo um utilizado com a parte onde está o texto e outro com a ilustração, como demonstra o código a seguir:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      xmlns:custom_text_view_font="http://schemas.android.com/apk/res-auto"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:background="@drawable/card_bg_shadow"
8      android:baselineAligned="false"
9      android:orientation="horizontal"
10     android:weightSum="100"
11     tools:ignore="NestedWeights,MergeRootFrame" >
12
13     <RelativeLayout
14         android:layout_width="0dip"
15         android:layout_height="match_parent"
16         android:layout_weight="40"
17         android:background="@null"
18         android:gravity="center"
19         android:weightSum="100" >
20
21         <include layout="@layout/book_text" />
22     </RelativeLayout>
23
24     <RelativeLayout
25         android:layout_width="0dip"
26         android:layout_height="match_parent"
27         android:layout_weight="60"
28         android:background="@null"

```

```

29         android:gravity="center"
30         android:weightSum="100" >
31
32         <include layout="@layout/book_image" />
33     </RelativeLayout>
34
35 </LinearLayout>

```

Foi criado um conjunto de atributos estáticos com as referências para os recursos áudio-visuais, como as imagens, textos, locuções e os layouts. Estes valores contém apenas o *id* dos recursos no projeto, como demonstra o código a seguir:

```

1  public interface BookResources {
2
3      public static final int[] BOOK_PAGES_TEXT = { R.raw.book_page_01,
4          R.raw.book_page_02, R.raw.book_page_03, R.raw.book_page_04,
5          R.raw.book_page_05, R.raw.book_page_06, R.raw.book_page_07,
6          BookResources.PAGE_WITHOUT_NARRATION };
7
8      public static final int BOOK_PAGES_SIZE = BOOK_PAGES_TEXT.length;
9
10     public static final int[] BOOK_PAGES_IMAGE = { R.drawable.page_1,
11         R.drawable.page_2, R.drawable.page_3, R.drawable.page_4,
12         R.drawable.page_5, R.drawable.page_6, R.drawable.page_7,
13         R.drawable.page_8 };
14
15     public static final int[] BOOK_PAGES_LAYOUT = { R.layout.book_simple_01,
16         R.layout.book_simple_02, R.layout.book_simple_03,
17         R.layout.book_simple_04, R.layout.book_simple_05,
18         R.layout.book_simple_06, R.layout.book_simple_07,
19         R.layout.book_simple_end };
20
21     public static final int PAGE_WITHOUT_NARRATION = -1;
22
23     public static final int[] audiosPath = { R.string.narration_page_1,
24         R.string.narration_page_2, R.string.narration_page_3,
25         R.string.narration_page_4, R.string.narration_page_5,
26         R.string.narration_page_6, R.string.narration_page_7,
27         BookResources.PAGE_WITHOUT_NARRATION };
28
29     public static final int LAYOUTS_LENGTH = BOOK_PAGES_LAYOUT.length;
30
31     public static final int AUDIOS_LENGTH = audiosPath.length;
32 }

```

As estruturas de dados são utilizadas na classe `BookController`, através dos métodos `createNarration()`, `createPageImage()`, `createPageText()` e inseridos no layout através dos métodos `loadPageImage()` e `loadPageText()`. Foi criado um objeto adapter com as páginas do livro, adicionando-as no objeto `ViewPager` que é o responsável pela transição entre as páginas, conforme demonstra o exemplo de código-fonte a seguir:

¹¹ **Classe `ViewPager`:** <https://developer.android.com/reference/android/support/v4/view/ViewPager>, acedido em 25/04/2019

```

1  public class BookController {
2
3      private ViewPager viewPager;
4
5      private ReducedCustomPagerAdapter adapter;
6
7      private int pageNumber;
8
9      public CharSequence pageNumberText;
10
11     private Context context;
12
13     private SparseArrayCompat<CharSequence> pageText = null;
14
15     private SparseArrayCompat<Drawable> pageImages = null;
16
17     private SparseArrayCompat<MediaPlayerManager> pageNarration = null;
18
19     public MenuItem menuItemPageNumber;
20
21     public static final String PERSISTENCE_KEY_PAGE = "PERSISTENCE_KEY_PAGE";
22
23     private LayoutInflater inflater;
24
25     public BookController(@NonNull final Context context) { ... }
26
27     private void createPageImage() {
28
29         final int size = BookResources.BOOK_PAGES_IMAGE.length;
30
31         if (pageImages == null) {
32
33             pageImages = new SparseArrayCompat<>(size);
34
35             final Resources res = context.getResources();
36
37             if (res != null) {
38                 for (int i = 0; i < size; i++) {
39
40                     final int pageIndexID = BookResources.BOOK_PAGES_IMAGE[i];
41
42                     if (pageIndexID != -1) {
43
44                         final Drawable imageDrawable = res.getDrawable(pageIndexID);
45
46                         if (imageDrawable != null) {
47                             pageImages.put(i, imageDrawable);
48                         } else {
49                             pageImages.put(i, null);
50                         }
51                     } else {
52                         pageImages.put(i, null);
53                     }
54                 }
55             }
56         }
57     }
58
59     private void createPageImageLow() { ... }

```

```

60
61 private void createPageText() {
62
63     if (pageText == null) {
64
65         final int pageLength = BookResources.BOOK_PAGES_TEXT.length;
66
67         pageText = new SparseArrayCompat<>(pageLength);
68
69         final int size = BookResources.BOOK_PAGES_TEXT.length;
70
71         final Resources res = context.getResources();
72
73         if (res != null) {
74
75             for (int i = 0; i < size; i++) {
76
77                 final int pageIndexID = BookResources.BOOK_PAGES_TEXT[i];
78
79                 final CharSequence page = GetTextRawFile.getTextRawFile(res,
80                     pageIndexID);
81
82                 if (page != null) {
83                     pageText.put(i, page);
84                 }
85             }
86         }
87     }
88
89 private void loadPageText() {
90
91     if (adapter != null && pageText != null && pageText.size() > 0) {
92
93         final int size = BookResources.BOOK_PAGES_LAYOUT.length;
94
95         for (int index = 0; i < size; index++) {
96
97             final View viewTemp = adapter.getViewByIndex(index);
98
99             if (viewTemp != null) {
100
101                 final JustifyTextViewCustomFont textViewPageText = viewTemp.
102                     findViewById(R.id.book_page_layout_text);
103
104                 if (textViewPageText != null && index < pageText.size()) {
105
106                     final CharSequence text = pageText.get(index);
107
108                     if (text != null && text.length() > 0) {
109                         textViewPageText.setTextJustify(text);
110                     }
111                 }
112             }
113         }
114     }
115
116 private void loadPageImage() {

```

```

117
118     if (adapter != null && pageImages != null && pageImages.size() > 0) {
119
120         final int size = BookResources.BOOK_PAGES_LAYOUT.length;
121
122         for (int index = 0; i < size; index++) {
123
124             final View viewTemp = adapter.getViewByIndex(index);
125
126             if (viewTemp != null) {
127
128                 final ImageView image = (ImageView) viewTemp.findViewById(R.id.
129                     book_page_layout_image);
130
131                 if (image != null) {
132
133                     final int pageImageSize = pageImages.size();
134
135                     if (pageImageSize > index) {
136                         final Drawable imageDrawable = pageImages.get(index);
137
138                         if (imageDrawable != null) {
139                             image.setImageDrawable(imageDrawable);
140                         }
141                     }
142                 }
143             }
144         }
145     }
146
147
148     private void createNarration() {
149
150         if (pageNarration == null) {
151
152             final int size = BookResources.AUDIOS_LENGTH;
153
154             pageNarration = new SparseArrayCompat<MediaPlayerManager>(size);
155
156             for (int i = 0; i < size; i++) {
157                 if (BookResources.audiosPath[i] == BookResources.PAGE_WITHOUT_NARRATION)
158                     {
159                         pageNarration.put(i, null);
160                     }
161                 else {
162                     final MediaPlayerManager mediaPlayerManager = getSoundMedia(
163                         BookResources.audiosPath[i]);
164                     if (mediaPlayerManager != null) {
165                         pageNarration.put(i, mediaPlayerManager);
166                     }
167                 }
168             }
169         }
170
171     private void createAdapter(final LayoutInflater inflater) { ... }
172
173     private void createViewPager() { ... }

```

```

173     private void start() { ... }
174
175     public final MediaPlayerManager getSoundMedia(final int resIdSoundMedia) { ... }
176
177     public final void startBookText(final View view) { ... }
178
179     private void goBookPage(final int page) { ... }
180
181     public final void bookPagePreviews() { ... }
182
183     public final void bookPageNext() { ... }
184
185     public final void pauseNarration() { ... }
186
187     public final void playNarration() { ... }
188
189     public final void resumeNarration() { ... }
190
191     public final void stopNarration() { ... }
192
193     public final void changePageNumberText() { ... }
194
195     public final void clearAll() { ... }
196
197     public final void savePage() { ... }
198
199     public final void loadPage() { ... }
200 }

```

4.3.3 Módulo – Livro Multimédia Pictogramas

Este módulo contém o livro multimédia com suporte para os símbolos pictográficos de comunicação com conversão de texto para voz, além dos recursos audiovisuais compostos por texto e imagem.

A interface é similar ao módulo de leitura básica e foi adicionado um campo com os pictogramas, estando localizado na parte superior de cada frase. Possui a opção de movimentação na vertical quando é ultrapassado o limite disponível para visualização, conforme demonstram as figuras 4.21 e 4.22.



Figura 4.21: Livro multimédia com os pictogramas I



Figura 4.22: Livro multimédia com os pictogramas II

Este módulo corresponde ao **RF06 – Livro multimédia com pictogramas**, é similar ao módulo anterior, as principais diferenças estão no layout que possui um componente customizado chamado `ImageViewAnimationTTS` que é baseado na classe `ImageView` mas implementa duas novas funcionalidades, uma com a animação do pictograma e outra com conversão de texto para voz. Um componente foi desenvolvido para permitir a rolagem na vertical e horizontal no mesmo layout, chama-se `LockableHorizontalScrollView`. As classes `LockableHorizontalScrollView` e `ImageViewAnimationTTS` serão explicadas mais adiante no módulo Framework. A seguir está um exemplo do código-fonte do layout utilizado com os pictogramas:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      xmlns:custom_text_view_font="http://schemas.android.com/apk/res-auto"
5      xmlns:imageview_animation="http://schemas.android.com/apk/res-auto"
6      android:layout_width="match_parent"
7      android:layout_height="wrap_content"
8      android:layout_marginBottom="@dimen/button_item_main_menu_padding"
9      android:layout_marginLeft="@dimen/button_item_main_menu_padding"
10     android:layout_marginTop="@dimen/button_item_main_menu_padding"
11     android:baselineAligned="false"
12     android:orientation="vertical" >
13
14     <com.toymobi.framework.view.scrollview.LockableHorizontalScrollView
15         android:layout_width="match_parent"
16         android:layout_height="match_parent"
17         android:background="@null"
18         android:clickable="true"
19         android:fadingEdge="none"
20         android:scrollbars="none" >
21
22         <LinearLayout
23             android:layout_width="wrap_content"
24             android:layout_height="wrap_content"
25             android:gravity="center_vertical"
26             android:orientation="horizontal" >
27
28             <com.toymobi.framework.view.imageview.ImageViewAnimationTTS
29                 android:id="@+id/imageViewSPC_1_1_1"
30                 android:layout_width="wrap_content"
31                 android:layout_height="wrap_content"
32                 android:padding="@dimen/button_item_spc_padding"
33                 android:src="@drawable/spc_eduardo"
34                 imageview_animation:layout="@anim/rotate_anti_clock_wise"
35                 imageview_animation:text="@string/spc_1_1_1" />
36
37             <com.toymobi.framework.view.imageview.ImageViewAnimationTTS
38                 android:id="@+id/imageViewSPC_1_1_2"
39                 android:layout_width="wrap_content"
40                 android:layout_height="wrap_content"
41                 android:padding="@dimen/button_item_spc_padding"
42                 android:src="@drawable/spc_era"
43                 imageview_animation:layout="@anim/rotate_anti_clock_wise"
44                 imageview_animation:text="@string/spc_1_1_2" />

```



```

45
46         (...)
47
48     </LinearLayout>
49 </com.toymobi.framework.view.scrollview.LockableHorizontalScrollView>
50
51 <com.toymobi.framework.view.textview.TextViewCustomFont
52     style="@style/TextBook"
53     android:text="@string/text_1_1"
54     android:textColor="@color/ACTION_BAR_COLOR"
55     custom_text_view_font:font="@string/roboto_black" />
56
57 </LinearLayout>

```

As classes implementadas neste módulo são similares às anteriores, as diferenças estão nas classes `Fragment` que carrega outros layouts e na classe `Controller` que contém os métodos para converter o texto para voz dos pictogramas. O trecho do código-fonte a seguir demonstram-se os métodos implementados:

```

1  public final void setTTS () {
2      if (imageViewSPC != null && imageViewSPC.size() > 0) {
3
4          final int length = imageViewSPC.size();
5
6          for (int i = 0; i < length; i++) {
7
8              final ImageViewAnimationTTS imageViewAnimationTTS = imageViewSPC.get(i);
9
10             if (imageViewAnimationTTS != null) {
11                 imageViewAnimationTTS.setTextToSpeech(textToSpeech);
12             }
13         }
14     }
15 }
16
17 private void stopTTS () {
18     if (EduardoStuff.ENABLE_TTS && textToSpeech != null) {
19         textToSpeech.stop();
20         textToSpeech.shutdown();
21     }
22 }
23
24 private void createItemsTTS () {
25     if (adapter != null) {
26         if (imageViewSPC == null) {
27
28             imageViewSPC = new ArrayList<ImageViewAnimationTTS>();
29
30             for (int i = 0; i < BookSPCResources.SPC_BUTTONS_PAGE_1_LENGTH; i++) {
31
32                 final ImageViewAnimationTTS temp = (ImageViewAnimationTTS) adapter.
33                     getViewByIndex(0).findViewById(BookSPCResources.
34                         SPC_BUTTONS_PAGE_1[i]);

```

```

34         if (temp != null) {
35             imageViewSPC.add(temp);
36         }
37     }
38
39     for (int i = 0; i < BookSPCResources.SPC_BUTTONS_PAGE_2_LENGTH; i++) {
40
41         final ImageViewAnimationTTS temp = (ImageViewAnimationTTS) adapter.
            getViewByIndex(1).findViewById(BookSPCResources.
                SPC_BUTTONS_PAGE_2[i]);
42
43         if (temp != null) {
44             imageViewSPC.add(temp);
45         }
46     }
47     (...)
48 }
49 }
50 }
51 }

```

Foi criado um layout para cada página que inclui todas as frases com os pictogramas, sendo inseridas dentro do objeto *adapter*, para facilitar o processo de conversão de texto para voz, todos os objetos foram inseridos numa estrutura de dados do tipo lista e depois inserido um *callback* quando o utilizador selecionar um determinado pictograma, conforme demonstra o trecho do código-fonte a seguir:

```

1  // CallBack dentro da classe Fragment para inicializar o TTS
2  public TextToSpeech textToSpeech;
3
4  @Override
5  public final View onCreateView (final LayoutInflater inflater,
6                                  final ViewGroup container,
7                                  final Bundle savedInstanceState) {
8
9      final View view = inflater.inflate(R.layout.layout_viewpager, container, false);
10
11      if (bookController == null) {
12
13          bookController = new BookSPCController(getActivity());
14
15          // usa um intent para checar se tts engine esta instalado
16          final Intent checkIntent = new Intent();
17          checkIntent.setAction(TextToSpeech.Engine.ACTION_CHECK_TTS_DATA);
18          startActivityForResult(checkIntent, MY_DATA_CHECK_CODE);
19
20      }
21
22      createActionBar();
23
24      if (view != null) {
25          bookController.startBookText(view);
26      }
27
28      return view;
29  }

```

```

30
31 @Override
32 public void onActivityResult (int requestCode, int resultCode, Intent data) {
33
34     if (requestCode == MY_DATA_CHECK_CODE &&
35         EduardoStuff.ENABLE_TTS) {
36
37         if (resultCode == TextToSpeech.Engine.CHECK_VOICE_DATA_PASS) {
38
39             final Context context = getContext();
40
41             bookController.textToSpeech = new TextToSpeech(context, BookSPCFrag.this);
42
43         } else {
44
45             final Intent installIntent = new Intent();
46             installIntent.setAction(TextToSpeech.Engine.ACTION_INSTALL_TTS_DATA);
47             startActivity(installIntent);
48
49         }
50     }
51 }
52
53 @Override
54 public void onInit (int status) {
55
56     if (EduardoStuff.ENABLE_TTS &&
57         status == TextToSpeech.SUCCESS &&
58         bookController != null) {
59
60         final Locale locateBR = new Locale("pt_BR");
61
62         final int result = bookController.textToSpeech.setLanguage(locateBR);
63
64         final Locale locatePT = new Locale("pt");
65
66         final int result_2 = bookController.textToSpeech.setLanguage(locatePT);
67
68         boolean checkPtTTS = result == TextToSpeech.LANG_MISSING_DATA ||
69                             result == TextToSpeech.LANG_NOT_SUPPORTED;
70
71         boolean checkBrTTS = result_2 == TextToSpeech.LANG_MISSING_DATA ||
72                             result_2 == TextToSpeech.LANG_NOT_SUPPORTED;
73
74         if (checkPtTTS || checkBrTTS) {
75             bookController.setTTS();
76         } else {
77             EduardoStuff.ENABLE_TTS = false;
78         }
79     } else {
80         EduardoStuff.ENABLE_TTS = false;
81     }
82 }

```

4.3.4 Módulo – Livro Multimédia Língua Gestual Portuguesa

Este módulo contém o livro multimédia com a tradução do texto para a língua gestual portuguesa. A interface é similar ao módulo de leitura básica, possui o mesmo mecanismo para a mudança de páginas. Foram removidos todos os efeitos sonoros e a locução neste módulo. Foi adicionado no lado esquerdo do layout um *player* para visualizar um vídeo, como demonstram as figuras 4.23 e 4.24.

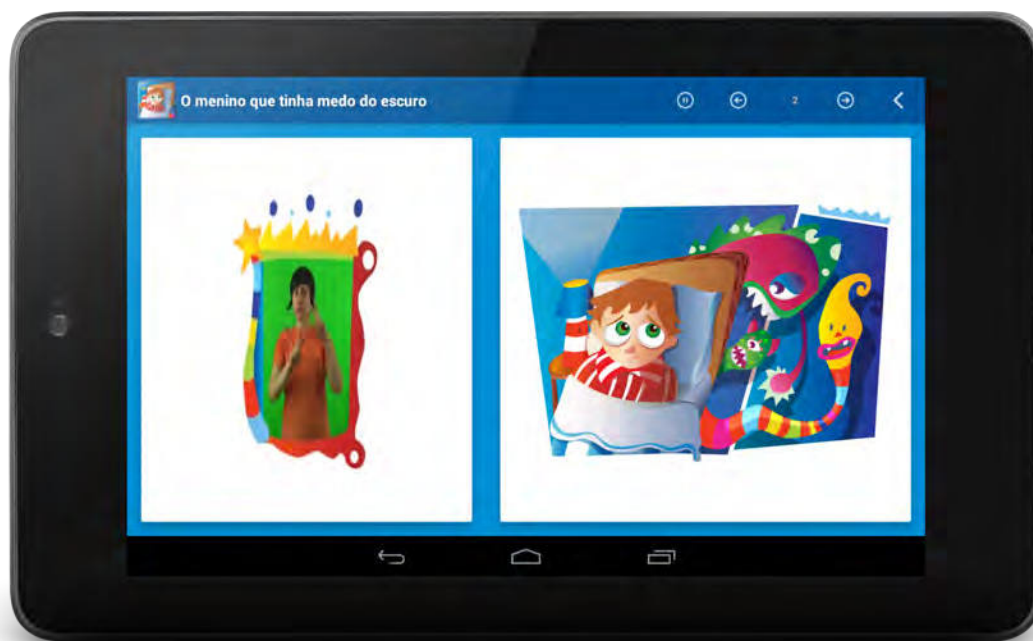


Figura 4.23: Livro multimédia com a língua gestual portuguesa I

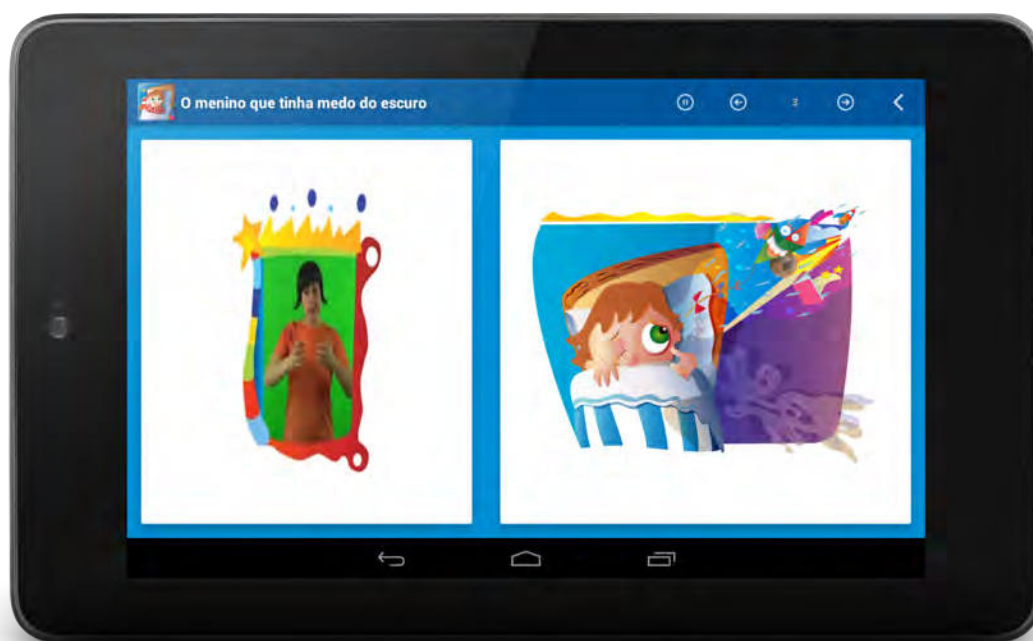


Figura 4.24: Livro multimédia com a língua gestual portuguesa II

Este módulo funciona como library e aplicação, corresponde com o **RF07 – Livro multimédia com a tradução para a língua gestual portuguesa**, é similar aos módulos do livro [RF05] e dos pictogramas [RF06], as principais diferenças estão no layout que inclui um player de vídeo e os métodos inseridos na classe Fragment para o controlo do vídeo, conforme demonstram os exemplos de código-fonte a seguir:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      android:background="@android:color/transparent"
7      android:keepScreenOn="true" >
8
9      <android.widget.VideoView
10         android:id="@+id/videoView"
11         android:layout_width="match_parent"
12         android:layout_height="match_parent"
13         android:layout_alignParentBottom="true"
14         android:layout_alignParentLeft="true"
15         android:layout_alignParentRight="true"
16         android:layout_alignParentTop="true"
17         android:layout_gravity="center"
18         android:background="@android:color/transparent" />
19
20 </RelativeLayout>

1  // Metodos adicionados na classe Fragment
2  private VideoView videoView;
3  private MediaController mediaController;
4
5  public final void createPlayer(@NonNull final View rootView) {
6
7      if (videoMediaControl == null) {
8          videoMediaControl = new VideoMediaControl();
9          videoView = (VideoView) rootView.findViewById(R.id.videoView);
10
11      if (videoView != null && context != null) {
12
13          mediaController = new MediaController(context) {
14              @Override
15              public void hide() {
16                  mediaController.show();
17              }
18
19              @Override
20              public void setAnchorView(final View view) { ... }
21          };
22
23          mediaController.setMediaPlayer(videoMediaControl);
24          videoView.setMediaController(mediaController);
25          videoView.requestFocus();
26      }
27      play();
28  }
29  }

```

```

30 public void play() {
31     if (context != null && videoView != null &&
32         videosPathString != null &&
33         videosPathString.size() > 0) {
34
35         videoView.setVideoURI(Uri.parse(videosPathString.get(pageNumber)));
36         videoView.start();
37
38         if (menuItemPlay != null) {
39             menuItemPlay.setIcon(R.drawable.icon_pause);
40             menuItemPlay.setTitle(R.string.pause_menu);
41             isPlayingVideo = true;
42         }
43
44         videoView.setOnCompletionListener(new OnCompletionListener() {
45             @Override
46             public void onCompletion(final MediaPlayer mediaPlayer) {
47
48                 if (mediaPlayer != null) {
49
50                     currentPosition = mediaPlayer.getDuration();
51
52                     if (menuItemPlay != null) {
53                         menuItemPlay.setIcon(R.drawable.icon_play);
54                         menuItemPlay.setTitle(R.string.play_menu);
55                         isPlayingVideo = false;
56                     }
57                 }
58             }
59         });
60         mediaController.show(0);
61     }
62 }
63
64 private void createVideosPath() {
65
66     SparseArrayCompat videosPathString = new SparseArrayCompat<String>(
67         BookGestualResources.BOOK_PAGES_SIZE);
68
69     final String path = "android.resource://" + context.getPackageName() + "/";
70
71     int i = 0;
72
73     for (final int id : BookGestualResources.BOOK_PAGES) {
74         videosPathString.put(i, path + id);
75         i++;
76     }
77 }
78
79 public final void resumeVideo() { ... }
80
81 public final void pauseVideo() { ... }
82
83 public final void stopVideo() { ... }

```

4.3.5 Módulo – Jogo Desenhar

Este módulo contém um jogo para desenhar, o layout foi dividido em quatro partes:

- Área principal onde é efetuada a pintura através do toque no ecrã.
- O campo inferior contém a área para seleccionar as cores.
- O campo inferior à direita contém a área para escolha do tipo de pincel e opção para apagar, de acordo com a dimensão escolhida pelo utilizador (pequeno, médio e grande).
- Área para seleccionar as opções de desenho com formas geométricas (ponto, quadrado, triângulo e círculo).

Na barra superior existe a opção de limpar toda a área da pintura e a cor desta barra é alterada de acordo com a seleção do utilizador, como demonstra a figura 4.25, 4.26 e 4.27.

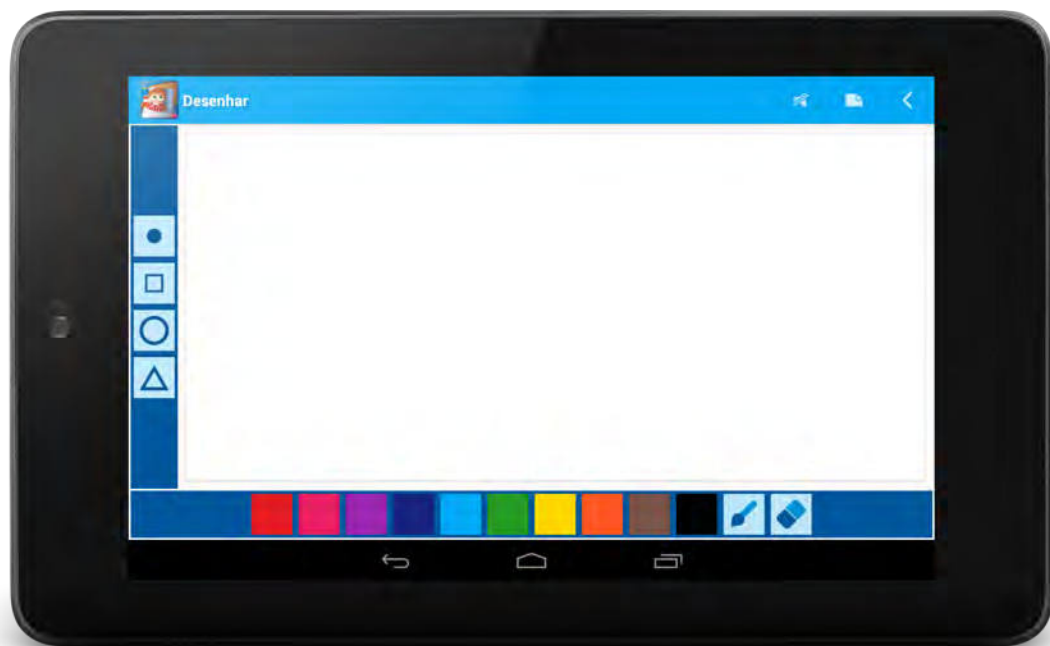


Figura 4.25: Jogo desenhar I



Figura 4.26: Jogo desenhar II



Figura 4.27: Jogo desenhar III

Este micro-serviço corresponde ao **RF08 – Jogo de desenhar**, seus principais elementos são o layout que utiliza um componente customizado chamado `DrawingView` sendo implementado para desenhar numa `View`¹² pontos e transformações geométricas, conforme demonstram os exemplos de código-fonte a seguir:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     android:layout_margin="2dp"
7     android:orientation="horizontal" >
8
9     <LinearLayout
10         android:layout_width="?attr/actionBarSize"
11         android:layout_height="match_parent"
12         android:layout_gravity="center"
13         android:background="@color/ACTION_BAR_COLOR"
14         android:gravity="center"
15         android:orientation="vertical" >
16
17         <ImageButton
18             android:id="@+id/button_dot"
19             android:layout_width="?attr/actionBarSize"
20             android:layout_height="?attr/actionBarSize"
21             android:background="@drawable/abc_item_background_holo_dark"
22             android:contentDescription="@null"
23             android:padding="4dp"
24             android:scaleType="fitXY"
25             android:src="@drawable/draw_game_icon_dot" />
26
27         <ImageButton
28             android:id="@+id/button_square"
29             android:layout_width="?attr/actionBarSize"
30             android:layout_height="?attr/actionBarSize"
31             android:background="@drawable/abc_item_background_holo_dark"
32             android:contentDescription="@null"
33             android:padding="4dp"
34             android:scaleType="fitXY"
35             android:src="@drawable/draw_game_icon_square" />
36
37         <ImageButton
38             android:id="@+id/button_circle"
39             android:layout_width="?attr/actionBarSize"
40             android:layout_height="?attr/actionBarSize"
41             android:background="@drawable/abc_item_background_holo_dark"
42             android:contentDescription="@null"
43             android:padding="4dp"
44             android:scaleType="fitXY"
45             android:src="@drawable/draw_game_icon_circle" />
46
47         <ImageButton
48             android:id="@+id/button_triangle"
49             android:layout_width="?attr/actionBarSize"
50             android:layout_height="?attr/actionBarSize"
```

¹² **Classe View:** <https://developer.android.com/reference/android/view/View>, acessado em 25/04/2019

```

51         android:background="@drawable/abc_item_background_holo_dark"
52         android:contentDescription="@null"
53         android:padding="4dp"
54         android:scaleType="fitXY"
55         android:src="@drawable/draw_game_icon_triangle" />
56     </LinearLayout>
57
58     <LinearLayout
59         android:layout_width="match_parent"
60         android:layout_height="match_parent"
61         android:background="@color/SNOW"
62         android:gravity="center_horizontal"
63         android:orientation="horizontal"
64         android:padding="8dp" >
65
66         <toymobi.eduardo.drawinggame.DrawingView
67             android:id="@+id/drawing"
68             android:layout_width="match_parent"
69             android:layout_height="match_parent"
70             android:background="@drawable/white_box" />
71     </LinearLayout>
72
73 </LinearLayout>

```

```

1  public class DrawingView extends View {
2      public static final int DRAW_DOT = 0;
3      public static final int DRAW_SQUARE = DRAW_DOT + 1;
4      public static final int DRAW_CIRCLE = DRAW_SQUARE + 1;
5      public static final int DRAW_TRIANGLE = DRAW_CIRCLE + 1;
6
7      private static final int SHAPE_STROKE_SIZE = 6;
8
9      public int drawType = DRAW_DOT;
10
11     private SimpleSoundPool sfx;
12
13     private Path drawPath, polygonTemp;
14
15     private Matrix matrix;
16
17     private Paint drawPaint, canvasPaint;
18
19     private int paintColor = 0xFFFFFFFF;
20
21     private Canvas drawCanvas;
22
23     private Bitmap canvasBitmap;
24
25     private float brushSize, lastBrushSize;
26
27     private boolean erase = false;
28
29     public DrawingView(final Context context) { ... }
30
31     private void init(@NonNull Context context) {
32
33         if (sfx == null) {
34             sfx = new SimpleSoundPool(context, R.raw.sfx_drawing);
35         }

```

```

36
37     brushSize = getResources().getInteger(R.integer.medium_size);
38     lastBrushSize = brushSize;
39     drawPath = new Path();
40     drawPaint = new Paint();
41     drawPaint.setColor(paintColor);
42     drawPaint.setAntiAlias(true);
43     drawPaint.setStrokeWidth(brushSize);
44     drawPaint.setStyle(Paint.Style.STROKE);
45     drawPaint.setStrokeJoin(Paint.Join.ROUND);
46     drawPaint.setStrokeCap(Paint.Cap.ROUND);
47     canvasPaint = new Paint(Paint.DITHER_FLAG);
48 }
49
50 @Override
51 public final boolean onTouchEvent(final MotionEvent event) {
52     final float touchX = event.getX();
53     final float touchY = event.getY();
54
55     switch (event.getAction()) {
56         case MotionEvent.ACTION_DOWN:
57             if (drawType == DRAW_DOT) {
58                 drawPath.moveTo(touchX, touchY);
59                 drawPath.lineTo(touchX + 1, touchY + 1);
60                 invalidate();
61             } else {
62                 drawShape(touchX, touchY);
63             }
64             return true;
65         case MotionEvent.ACTION_MOVE:
66             if (drawType == DRAW_DOT) {
67                 drawPath.lineTo(touchX, touchY);
68                 invalidate();
69             } else {
70                 drawShape(touchX, touchY);
71             }
72
73             return true;
74         case MotionEvent.ACTION_UP:
75             if (sfx != null && GlobalSettings.soundEnable) {
76                 sfx.playSound(R.raw.sfx_drawing);
77             }
78
79             if (drawType == DRAW_DOT) {
80                 drawShape(touchX, touchY);
81             }
82             return true;
83         default:
84             return super.onTouchEvent(event);
85     }
86 }
87
88 private void drawShape(final float touchX, final float touchY) {
89     switch (drawType) {
90
91         case DRAW_DOT:
92             drawCanvas.drawPath(drawPath, drawPaint);
93             drawPath.reset();
94             break;

```

```

95
96         case DRAW_SQUARE:
97             drawPaint.setStrokeWidth(SHAPE_STROKE_SIZE);
98             drawPath.addPath(getPoligon(touchX, touchY, (int) brushSize, 4, 45f));
99             drawCanvas.drawPath(drawPath, drawPaint);
100             drawPath.reset();
101             break;
102
103         case DRAW_CIRCLE:
104             drawPaint.setStrokeWidth(SHAPE_STROKE_SIZE);
105             drawPath.addCircle(touchX, touchY, brushSize, Path.Direction.CCW);
106             drawCanvas.drawPath(drawPath, drawPaint);
107             drawPath.reset();
108             break;
109
110         case DRAW_TRIANGLE:
111             drawPaint.setStrokeWidth(SHAPE_STROKE_SIZE);
112             drawPath.addPath(getPoligon(touchX, touchY, (int) brushSize, 3, 30f));
113             drawCanvas.drawPath(drawPath, drawPaint);
114             drawPath.reset();
115             break;
116     }
117     invalidate();
118 }
119
120 @Override
121 protected final void onSizeChanged(final int width, final int heighth,
122                                     final int oldWidth, final int oldHeighth) { ... }
123
124 @Override
125 protected final void onDraw(final Canvas canvas) { ... }
126
127 public final void setColor(final int newColor) { ... }
128
129 public final void setBrushSize(final float newSize) { ... }
130
131 public final void setLastBrushSize(final float lastSize) { ... }
132
133 public final float getLastBrushSize() { ... }
134
135 public final void setErase(final boolean isErase) { ... }
136
137 public final void startNewPaper() { ... }
138
139 private Path getPoligon(final float centerX, final float centerY,
140                         final int radius, final int numberPoints,
141                         final float degrees) { ... }
142 }

```

4.3.6 Módulo – Jogo da Memória

Este módulo contém um jogo de cartas que funciona através da seleção de um par com as figuras iguais. Contém os níveis de dificuldade com base na quantidade de cartas (6, 12 e 18) e com o tempo de visualização inicial. Quando um par de cartas é selecionado faz-se a verificação das figuras, se forem iguais permanecem visíveis até que o último par seja selecionado, caso contrário fecham-se as cartas para o estado inicial. Na barra superior existe uma opção para alterar quantidade de cartas no jogo, como demonstram as figuras 4.28, 4.29 e 4.30.

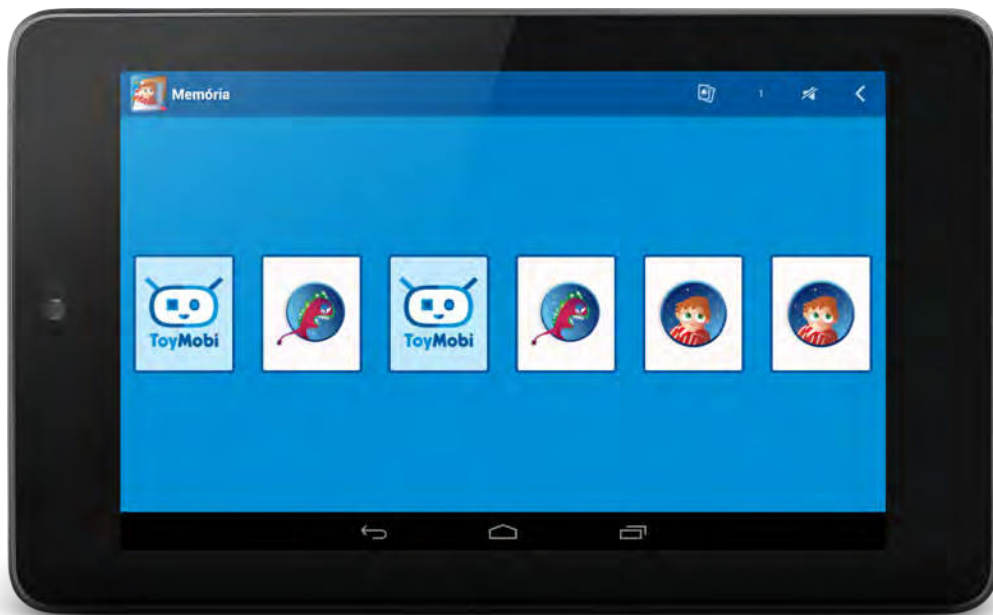


Figura 4.28: Jogo da memória I

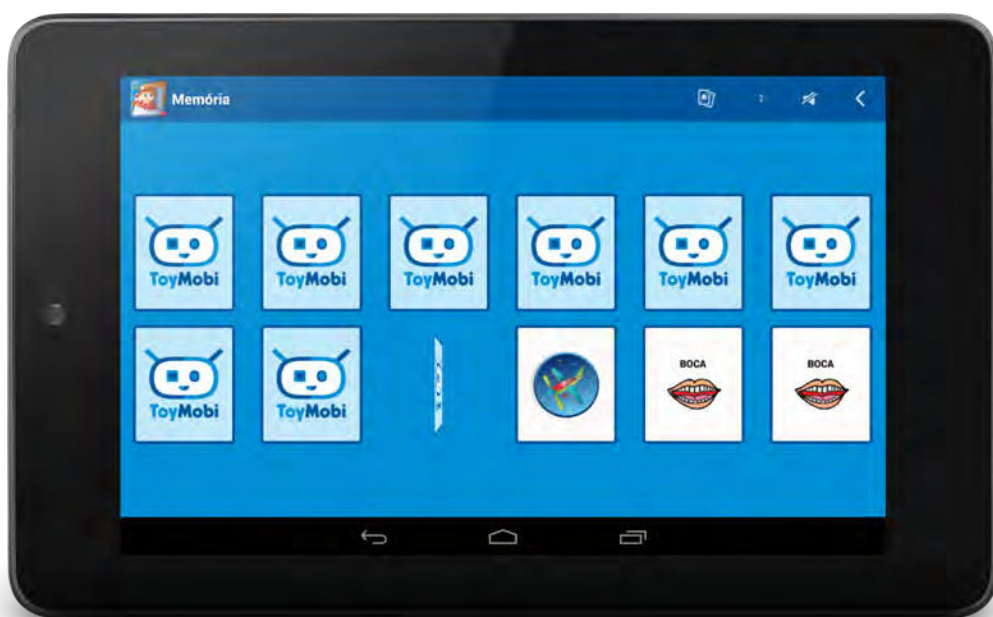


Figura 4.29: Jogo da memória II

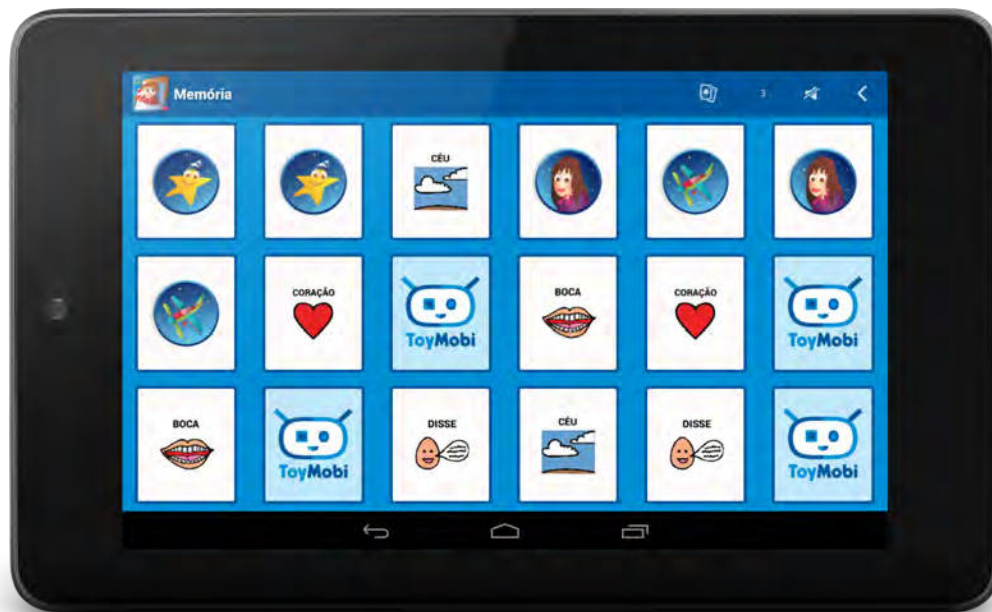


Figura 4.30: Jogo da memória III

Este micro-serviço corresponde ao **RF09 – Jogo de memória**, sendo uma das principais classes desenvolvidas neste módulo a classe Card. Possui um layout próprio, um Listener com a ação de abrir e fechar a carta além de seleção da carta, conforme demonstra exemplo do código-fonte a seguir:

```

1 public class Card {
2
3     private OnCheckedChangeListener onCheckedChangeListener;
4
5     public int number, index;
6
7     private boolean ignoreCheckStateCard = false;
8
9     private ImageView front, back;
10
11     private Bitmap backgroundImageBack;
12
13     public ViewAnimator viewAnimator;
14
15     private OnClickListener openClickListener;
16
17     private VibrationFeedback vibrationFeedback;
18
19     private ViewGroup VIEW_GROUP_EMPTY;
20
21     public static boolean blockedCards = false;
22
23     public Card (@NonNull final Context context, LayoutInflater inflater,
24                 final int back_card_res_image_id, final int index,
25                 final OnCheckedChangeListener onCheckedChangeListener) {
26
27         ( ... )
28

```

```

29         viewAnimator = (ViewAnimator) LayoutInflater.from(context).
30             inflate(R.layout.card_cell, VIEW_GROUP_EMPTY);
31
32         if (viewAnimator != null) {
33
34             viewAnimator.setOnClickListener(new OnClickListener() {
35                 @Override
36                 public void onClick (final View view) {
37                     AnimationFactory.flipTransition(viewAnimator, FlipDirection.
38                         LEFT_RIGHT);
39                 }
40             });
41
42             createListener();
43
44             front = (ImageView) viewAnimator.findViewById(R.id.front);
45             if (front != null) {
46                 front.setOnClickListener(openClickListener);
47             }
48
49             back = (ImageView) viewAnimator.findViewById(R.id.back);
50
51             ( ... )
52         }
53
54     private void createListener () {
55         openClickListener = new OnClickListener() {
56             @Override
57             public void onClick (final View view) {
58                 if (!blockedCards) {
59
60                     AnimationFactory.flipTransition(viewAnimator, FlipDirection.
61                         LEFT_RIGHT);
62
63                     if (!ignoreCheckStateCard) {
64                         onCheckStateListener.onCheckState(index);
65                     }
66                 }
67             };
68         }
69
70     public void close () { ... }
71
72     public void disableClick () { ... }
73
74     public void enableClick () { ... }
75
76     public void cleanAll () { ... }
77
78     public void open_card () { ... }
79
80     public void close_card () { ... }
81
82     private void deallocate () { ... }
83 }

```

A classe implementada para definir os níveis de dificuldade de acordo com o tempo de exibição e a quantidade de cartas disponíveis foi a CardGameIA, conforme demonstra o código-fonte a seguir:

```
1 public class CardGameIA {
2
3     public static enum SPEED_STATE {
4         SPEED_SLOW, SPEED_NORMAL, SPEED_FAST;
5     }
6
7     public static enum NUMBER_CARDS_STATE {
8         CARDS_6, CARDS_12, CARDS_18;
9     }
10
11     public static final int MAX_LEVELS = 8;
12
13     public int cardsCorrect = 0, currentLevel = 0;
14
15     public static SparseArrayCompat<LevelType> levels;
16
17     public CardGameIA () {
18         createLevel();
19     }
20
21     private void createLevel () {
22         if (levels == null) {
23             levels = new SparseArrayCompat<LevelType>(MAX_LEVELS);
24
25             levels.put(0, new LevelType(SPEED_STATE.SPEED_SLOW, NUMBER_CARDS_STATE.
26                 CARDS_6));
27
28             levels.put(1, new LevelType(SPEED_STATE.SPEED_NORMAL, NUMBER_CARDS_STATE.
29                 CARDS_6));
30
31             levels.put(2, new LevelType(SPEED_STATE.SPEED_FAST, NUMBER_CARDS_STATE.
32                 CARDS_6));
33
34             levels.put(3, new LevelType(SPEED_STATE.SPEED_SLOW, NUMBER_CARDS_STATE.
35                 CARDS_12));
36
37             levels.put(4, new LevelType(SPEED_STATE.SPEED_NORMAL, NUMBER_CARDS_STATE.
38                 CARDS_12));
39
40             levels.put(5, new LevelType(SPEED_STATE.SPEED_FAST, NUMBER_CARDS_STATE.
41                 CARDS_12));
42
43             levels.put(6, new LevelType(SPEED_STATE.SPEED_SLOW, NUMBER_CARDS_STATE.
44                 CARDS_18));
45
46             levels.put(7, new LevelType(SPEED_STATE.SPEED_NORMAL, NUMBER_CARDS_STATE.
47                 CARDS_18));
48
49             levels.put(8, new LevelType(SPEED_STATE.SPEED_FAST, NUMBER_CARDS_STATE.
50                 CARDS_18));
51         }
52     }
53 }
```



```

45     public final SPEED_STATE cardRotateSpeed () { ... }
46
47     public final void setCardRotateSpeed (SPEED_STATE speedState) { ... }
48
49     public final void setCardColumnNumber (NUMBER_CARDS_STATE cardColumnNumber) { ... }
50
51     public final NUMBER_CARDS_STATE cardColumnNumber () { ... }
52 }

```

A classe Controller deste módulo tem como principais métodos um Handler que é responsável para criar uma máquina de estado no jogo e um método para verificar a condição de acerto (quando é selecionada um par de cartas) e a vitória quando todos os pares forem selecionados, conforme demonstra o trecho do código-fonte a seguir:

```

1  public Handler cardsStateHandler = new Handler() {
2      public void handleMessage (final Message message) {
3          if (message != null) {
4              final int responde_message = message.what;
5
6              switch (responde_message) {
7
8                  case CardValues.RESET_CARDS_STATE:
9                      reset();
10                     break;
11
12                 case CardValues.DISABLE_ALL_CARDS_STATE:
13                     disableAllCardsState();
14                     break;
15
16                 case CardValues.ENABLE_ALL_CARDS_STATE:
17                     enableAllCardsState();
18                     break;
19
20                 case CardValues.OPEN_CARDS_STATE:
21                     openCardsState();
22                     break;
23
24                 case CardValues.CLOSE_CARDS_STATE:
25                     closeCardsState();
26                     break;
27
28                 case CardValues.OPEN_ALL_CARDS_STATE:
29                     openAllCardsState();
30                     break;
31
32                 case CardValues.CLOSE_ALL_CARDS_STATE:
33                     closeAllCardsState();
34                     break;
35
36                 case CardValues.REMOVE_CALLBACK_STATE:
37                     this.removeCallbacksAndMessages(null);
38                     break;
39             }
40         }
41     }
42 };

```

```

43
44 public void checkCards (final int number_card_1, final int number_card_2,
45                         final int index_card_1, final int index_card_2) {
46
47     if (CardValues.cardsStates[index_card_1] != CardValues.CARD_STATE.CORRECT
48         && CardValues.cardsStates[index_card_2] != CardValues.CARD_STATE.CORRECT) {
49
50         if (number_card_1 == number_card_2) {
51
52             CardValues.cardsStates[index_card_1] = CardValues.CARD_STATE.CORRECT;
53             CardValues.cardsStates[index_card_2] = CardValues.CARD_STATE.CORRECT;
54             cardGameIA.cardsCorrect++;
55
56             if (cardGameIA.cardsCorrect == CardValues.maxCells >> 1) {
57                 win();
58             }
59
60         } else {
61
62             final Card card1 = cards.get(index_card_1);
63
64             if (card1 != null) {
65                 CardValues.cardsStates[index_card_1] = CardValues.CARD_STATE.CLOSED;
66                 createCardRunnable(card1);
67             }
68
69             final Card card2 = cards.get(index_card_2);
70
71             if (card2 != null) {
72                 CardValues.cardsStates[index_card_2] = CardValues.CARD_STATE.CLOSED;
73                 createCardRunnable(card2);
74             }
75         }
76         CardValues.state_card = CardValues.SELECT_CARD_1_STATE;
77     }
78     enableAction();
79 }
80
81 public void createCardRunnable (final Card card) {
82     if (card != null) {
83
84         if (cardHandler == null) {
85             cardHandler = new Handler();
86         }
87
88         cardRunnable = new Runnable() {
89             @Override
90             public void run () {
91                 card.close();
92             }
93         };
94         cardHandler.postDelayed(cardRunnable, CardValues.CARD_ANIMATION_TIME);
95     }
96 }

```

4.3.7 Módulo – Jogo do Puzzle

Este módulo contém o jogo de quebra-cabeça (puzzle), o objetivo é movimentar as peças colocando-as na posição correta. Ao inserir todas as peças nas posições corretas uma nova imagem é apresentada como desafio. O layout é composto pela ilustração dividida em 18 partes que são espalhadas aleatoriamente no campo principal, a movimentação é realizada ao arrastar uma peça para a posição desejada. Na barra superior existe a opção para alterar a ilustração, como demonstram as figuras 4.31, 4.32 e 4.33.



Figura 4.31: Jogo de quebra-cabeça I



Figura 4.32: Jogo de quebra-cabeça II



Figura 4.33: Jogo de quebra-cabeça III

Este micro-serviço corresponde ao **RF10 – Jogo de quebra-cabeça**, sendo os principais elementos deste módulo uma classe chamada `PuzzleGridView`, que é baseada na classe `ViewGroup`¹³. A implementação cria um grid com as imagens da ilustração divididas uniformemente no ecrã principal. A seguir está um trecho do código-fonte desta classe e do Controller para demonstração:

```

1  public class PuzzleGridView extends ViewGroup implements View.OnTouchListener {
2
3      private static final int ANIMATION_TIME = 150;
4
5      private static final int START_POINT_VALUE = -1;
6
7      private int column, view_size;
8
9      private static final int padding = 1;
10
11     private float lastDeltaTime = 0;
12
13     private Handler handler = new Handler();
14
15     private int dragged = START_POINT_VALUE, lastX = START_POINT_VALUE,
16         lastY = START_POINT_VALUE, lastTarget START_POINT_VALUE;
17
18     private boolean isTouching = false;
19
20     private List<Integer> newPositionList;
21
22     private OnPuzzleReorderListener onRearrangeListener;
23
24     private SimpleSoundPool sfx;

```

¹³ Classe `ViewGroup`: <https://developer.android.com/reference/android/view/ViewGroup>, acedido em 25/04/2019

```

25
26     public PuzzleGridView (final Context context, final int view_size, final int column)
27         { ... }
28
29     private Runnable updateTask = new Runnable() { ... }
30
31     @Override
32     public void addView (final View child) { ... }
33
34     @Override
35     public void removeViewAt (final int index) { ... }
36
37     @Override
38     protected void onLayout (final boolean changed, final int l, final int t, final int
39         r, final int b) { ... }
40
41     @Override
42     protected int getChildDrawingOrder (final int childCount, final int i) { ... }
43
44     private int getIndexFromCoordinate (final int x, final int y) { ... }
45
46     private int getTargetFromCoordinate (int target_coordinate) { ... }
47
48     private int getTargetFromCoordinate (final int x, final int y) { ... }
49
50     private void animateDragged () { ... }
51
52     private void animateGap (final int target) { ... }
53
54     private void reorderChildren () { ... }
55
56     private int getLastIndex () { ... }
57
58     public void setOnRearrangeListener (final OnPuzzleReorderListener l){ ... }
59
60     public final void deallocate () { ... }
61 }

```

```

1  public class PuzzleController {
2
3      private Random random;
4
5      private PuzzleGridView puzzleGridView;
6
7      public int width, height, view_size, index_image;
8
9      private Context context;
10
11     private final List<CharSequence> new_goal = new ArrayList<CharSequence>(PuzzleValues.
12         SIZE);
13
14     private final List<CharSequence> poem = new ArrayList<CharSequence>(PuzzleValues.SIZE);
15
16     private final List<CharSequence> result_ok = new ArrayList<CharSequence>(PuzzleValues.
17         SIZE);
18
19     private SparseArrayCompat<BitmapDrawable> imagesList;
20
21     private SparseArrayCompat<Bitmap> piecesImageList;

```

```

20
21 private ViewGroup container;
22
23 private int textSize, textColor;
24
25 public MenuItem menuItemChangePicture;
26
27 private SimpleSoundPool sfx;
28
29 public PuzzleController(@NonNull Context context, @NonNull ViewGroup layout_container) {
30
31     this.context = context;
32
33     this.container = layout_container;
34
35     int widthTemp = DeviceDimensionsUtil.getDisplayWidth(context);
36
37     if (!EduardoStuff.isNormalSize(context)) {
38
39         final int h = DeviceDimensionsUtil.getDisplayHeight(context);
40
41         widthTemp = h + (h / 4);
42     }
43
44     view_size = widthTemp / PuzzleValues.COLUMN;
45
46     width = view_size * PuzzleValues.COLUMN;
47
48     height = view_size * PuzzleValues.LINE;
49
50     index_image = 0;
51
52     ( ... )
53 }
54
55 private void start() { ... }
56
57 private void loadBkgImages() {
58
59     if (context != null) {
60         final Resources resources = context.getResources();
61
62         if (resources != null) {
63
64             final int imageLength = EduardoStuff.isNormalSize(context) ?
65                 PuzzleValues.IMAGES_RES_ID.length :
66                 PuzzleValues.IMAGES_RES_ID_SMALL.length;
67
68             if (imagesList == null) {
69
70                 imagesList = new SparseArrayCompat<BitmapDrawable>(imageLength);
71
72                 BitmapDrawable bkg_image_temp = null;
73
74                 int imageSizeRes = EduardoStuff.isNormalSize(context) ?
75                     PuzzleValues.IMAGES_RES_ID[i] :
76                     PuzzleValues.IMAGES_RES_ID_SMALL[i];
77
78

```

```

79         for (int i = 0; i < imageLength; i++) {
80
81             Bitmap bitmap = UIComponentUtils.
82                 decodeSampledBitmapFromResource(resources,
83                                                     imageSizeRes,
84                                                     width, height)
85
86             if (bitmap != null) {
87                 bkg_image_temp = new BitmapDrawable(resources, bitmap);
88
89                 imagesList.put(i, bkg_image_temp);
90                 bkg_image_temp = null;
91             }
92         }
93     }
94 }
95
96 }
97
98 private void createBkgImages() {
99
100     if (piecesImageList == null) {
101         piecesImageList = new SparseArrayCompat<Bitmap>(PuzzleValues.SIZE);
102     }
103
104     Bitmap croc_bitmap = null;
105
106     Bitmap bitmapTemp = imagesList.get(index_image).getBitmap();
107
108     for (int i = 0; i < PuzzleValues.SIZE; i++) {
109
110         final int line = i / PuzzleValues.COLUMN;
111
112         final int column = ((i < PuzzleValues.COLUMN) ? i :
113                             (i % (line * PuzzleValues.COLUMN)));
114
115         final int x = view_size * column;
116
117         final int y = view_size * line;
118
119         if (imagesList != null) {
120
121             croc_bitmap = Bitmap.createBitmap(bitmapTemp, x, y, view_size, view_size);
122
123             if (croc_bitmap != null) {
124
125                 piecesImageList.put(i, croc_bitmap);
126
127                 croc_bitmap = null;
128             }
129         }
130     }
131 }
132
133 private void addPuzzle() { ... }
134
135 private void setListeners() { ... }
136
137 private void addDraggableGridView() { ... }

```

```

138
139 private void win() { ... }
140
141 private Bitmap drawTextToBitmap(final CharSequence textNumber) { ... }
142
143 private void clean() { ... }
144
145 public void deallocateAll() { ... }
146
147 private void showMessageWinner() { ... }
148
149 private void loadSfx() { ... }
150
151 public final void changeLevel() { ... }
152 }

```

4.3.8 Módulo – Jogo Completar as Palavras

Este jogo tem o objetivo de completar uma palavra com as letras corretas. O layout é composto por duas áreas, no campo principal está a palavra incompleta e as letras, no secundário está uma figura que representa a palavra com opção de converter o texto em voz. A movimentação é realizada ao arrastar uma das três opções até preencher a posição vazia. Quando o utilizador inserir a letra correta uma nova palavra é apresentada, caso contrário a mesma retorna para a sua posição de origem, como demonstram as figuras 4.34, 4.35 e 4.36.



Figura 4.34: Jogo de completar as palavras I



Figura 4.35: Jogo de completar as palavras II



Figura 4.36: Jogo de completar as palavras III

Este micro-serviço corresponde ao **RF11 – Jogo para completar a palavra**. As principais classes desenvolvidas neste módulo foram a `DragAndDropGameView` que é um layout customizado baseado na classe `RelativeLayout`¹⁴ com a funcionalidade de mover um item no ecrã e verificar se está no local correto, a `DragItem` que é uma classe que faz a leitura e inserção da palavra neste jogo, além do `Controller`. A seguir estão os exemplos do código-fonte destas classes:

```

1 public class DragAndDropGameView extends RelativeLayout implements onTouchListener {
2
3     public static final int[] SPOTS_RES_IDS = {R.id.spot1, R.id.spot2,
4         R.id.spot3, R.id.spot4, R.id.spot5, R.id.spot6, R.id.spot7,
5         R.id.spot8, R.id.spot9};
6
7     public static final int MAX_LETTER_SPOT = SPOTS_RES_IDS.length;
8
9     private static final int OFFSET_TOUCH_POSITIVE = 32;
10
11     private static final int OFFSET_TOUCH_NEGATIVE = -32;
12
13     private SparseArrayCompat<ImageView> spotItems = null;
14
15     private View selectedItemView = null;
16
17     private int offsetX = 0, offsetY = 0;
18
19     private boolean isTouch = false, isDrop = false;
20
21     private android.view.ViewGroup.LayoutParams imageParams,
22         imageParamsLetter1, imageParamsLetter2, imageParamsLetter3;
23
24     private ImageView letter1, letter2, letter3;
25
26     private int crashX, crashY;
27
28     private int topY, leftX, rightX, bottomY;
29
30     private int indexSpot = 0;
31
32     private CharSequence checkCurrentLetterTag;
33
34     private DragItem dragItem;
35
36     public OnNextItemListener onNextItemListener;
37
38     public DragAndDropGameView(final Context context) { ... }
39
40     public DragAndDropGameView(final Context context, final AttributeSet attrs) { ... }
41
42     public DragAndDropGameView(final Context context,
43         final AttributeSet attrs, final int defStyle) { ... }
44
45     @Override
46     protected void onFinishInflate() { ... }

```

¹⁴ **Classe RelativeLayout:** <https://developer.android.com/reference/android/widget/RelativeLayout>, acedido em 25/04/2019

```

47
48     private void init(@NonNull Context context) {
49
50         inflate(context, R.layout.drag_and_drop_layout, this);
51
52         if (spotItems == null) {
53
54             final int lenght = SPOTS_RES_IDS.length;
55
56             spotItems = new SparseArrayCompat<ImageView>(lenght);
57
58             for (int i = 0; i < lenght; i++) {
59                 final ImageView spot = (ImageView) findViewById(SPOTS_RES_IDS[i]);
60                 if (spot != null) {
61                     spotItems.put(i, spot);
62                 }
63             }
64         }
65
66         final ViewGroup container = (ViewGroup) findViewById(R.id.container);
67
68         letter1 = (ImageView) findViewById(R.id.letter_1);
69
70         letter2 = (ImageView) findViewById(R.id.letter_2);
71
72         letter3 = (ImageView) findViewById(R.id.letter_3);
73
74         final RelativeLayout.LayoutParams letterDragLayoutParams = new RelativeLayout.
75             LayoutParams(
76                 new ViewGroup.MarginLayoutParams(
77                     RelativeLayout.LayoutParams.WRAP_CONTENT,
78                     RelativeLayout.LayoutParams.WRAP_CONTENT));
79
80         container.measure(MeasureSpec.UNSPECIFIED, MeasureSpec.UNSPECIFIED);
81
82         final int widthContainer = DeviceDimensionsUtil.getDisplayWidth(context);
83
84         final int heightContainer = DeviceDimensionsUtil
85             .getDisplayHeight(context);
86
87         final Rect rect = new Rect(OFFSET_TOUCH_NEGATIVE,
88             OFFSET_TOUCH_NEGATIVE, widthContainer
89             - OFFSET_TOUCH_POSITIVE, heightContainer
90             - OFFSET_TOUCH_POSITIVE);
91
92         container.setOnTouchListener(new View.OnTouchListener() {
93
94             public boolean onTouch(final View view, final MotionEvent event) {
95
96                 if (isTouch) {
97
98                     final ImageView spotByIndex = spotItems.get(indexSpot);
99
100                     switch (event.getActionMasked()) {
101
102                         case MotionEvent.ACTION_DOWN:
103
104                             if (spotByIndex != null) {

```

```

105         topY = spotByIndex.getTop();
106         leftX = spotByIndex.getLeft();
107         rightX = spotByIndex.getRight();
108         bottomY = spotByIndex.getBottom();
109     }
110
111     break;
112
113     case MotionEvent.ACTION_MOVE:
114         crashX = (int) event.getX();
115         crashY = (int) event.getY();
116
117         final int x = (int) event.getX() - offsetX;
118         final int y = (int) event.getY() - offsetY;
119
120         letterDragLayoutParams.setMargins(x, y, 0, 0);
121
122         // Drop Image Here
123         if (crashX > leftX && crashX < rightX
124             && crashY > topY && crashY < bottomY) {
125
126             checkCurrentLetterTag = (CharSequence) selectedItemView
127                 .getTag();
128
129             if (checkCurrentLetterTag
130                 .equals(DragItem.CORRECT)) {
131                 if (dragItem != null) {
132                     if (spotByIndex != null) {
133
134                         spotByIndex
135                             .setImageResource(dragItem.
136                                 letterImageCorrect);
137
138                         if (onNextItemListener != null) {
139                             onNextItemListener.nextItem();
140                         }
141                     }
142                     isDrop = true;
143
144                     if (selectedItemView != null) {
145                         selectedItemView
146                             .setVisibility(View.INVISIBLE);
147                     }
148                 } else {
149                     event.setAction(MotionEvent.ACTION_UP);
150                     onTouch(view, event);
151                 }
152             } else {
153                 if (selectedItemView != null) {
154                     // Drop Image Here
155                     if (rect != null && rect.contains(x, y)) {
156                         selectedItemView
157                             .setLayoutParams(letterDragLayoutParams)
158                             ;
159                     } else {
160                         event.setAction(MotionEvent.ACTION_UP);
161                         onTouch(view, event);
162                     }
163                 }
164             }
165         }
166     }

```

```

162         }
163     }
164     break;
165     case MotionEvent.ACTION_UP:
166         isTouch = false;
167         if (isDrop) {
168             isDrop = false;
169         } else {
170             selectedItemView.setLayoutParams(imageParams);
171         }
172         break;
173     }
174 }
175 return true;
176 }
177 });
178
179 letter1.setOnTouchListener(this);
180 letter2.setOnTouchListener(this);
181 letter3.setOnTouchListener(this);
182
183 imageParamsLetter1 = letter1.getLayoutParams();
184 imageParamsLetter2 = letter2.getLayoutParams();
185 imageParamsLetter3 = letter3.getLayoutParams();
186 }
187
188 @Override
189 public boolean onTouch(final View view, final MotionEvent event) { ... }
190
191 public final void deallocate() { ... }
192
193 public final void setDragItem(final DragItem dragItem) { ... }
194
195 private void setAttributesIA() { ... }
196
197 public final void eraserSpot() { ... }
198
199 public final void start() { ... }

```



```

1 public class DragItem {
2
3     public static final CharSequence CORRECT = "correct";
4     public static final CharSequence INCORRECT = "incorrect";
5
6     public static final int LETTER_1 = 0;
7     public static final int LETTER_2 = 1;
8     public static final int LETTER_3 = 2;
9
10    private static final int MAX_LETTER_SPOT = 3;
11
12    public CharSequence name;
13
14    public Drawable image;
15
16    public int lengthName;
17
18    public int indexGoalSpotUp = 0, indexGoalSpotDown = 0;
19
20    private static Random random;

```

```

21
22     public SparseArrayCompat<ItemLetter> itemsLetter;
23
24     public int letterImageCorrect = 0,
25         letterImageIncorrectCorrect1 = 0,
26         letterImageIncorrectCorrect2 = 0;
27
28     public DragItem(@NonNull Resources resources, final int nameId, final int imageId) {
29
30         name = resources.getText(nameId);
31
32         image = resources.getDrawable(imageId);
33
34         if (name != null) {
35
36             lengthName = name.length();
37
38             if (itemsLetter == null) {
39                 itemsLetter = new SparseArrayCompat<ItemLetter>(lengthName);
40             }
41
42             for (int i = 0; i < lengthName; i++) {
43
44                 final char charLetter = name.charAt(i);
45
46                 final int imageLetterId = AlphabetStruct.getImageLetter(
47                     charLetter, resources);
48
49                 final ItemLetter item = new ItemLetter(charLetter,
50                     imageLetterId);
51
52                 itemsLetter.put(i, item);
53             }
54             init();
55         }
56     }
57
58     public final void init() {
59         if (random == null) {
60             random = new Random();
61         }
62
63         if (lengthName > 0 && lengthName <= DragAndDropGameView.MAX_LETTER_SPOT) {
64
65             indexGoalSpotUp = random.nextInt(lengthName);
66
67             indexGoalSpotDown = random.nextInt(MAX_LETTER_SPOT);
68
69             letterImageCorrect = itemsLetter.get(indexGoalSpotUp).letterImageId;
70
71             boolean finishRandom = false;
72
73             while (!finishRandom) {
74
75                 letterImageIncorrectCorrect1 = AlphabetStruct.alphabet_image_id[random
76                     .nextInt(AlphabetStruct.alphabet_image_id.length)];
77
78                 letterImageIncorrectCorrect2 = AlphabetStruct.alphabet_image_id[random
79                     .nextInt(AlphabetStruct.alphabet_image_id.length)];

```

```

80
81         if (letterImageIncorrectCorrect1 != letterImageCorrect
82             && letterImageIncorrectCorrect2 != letterImageCorrect
83             && letterImageIncorrectCorrect1 != letterImageIncorrectCorrect2)
84             {
85                 finishRandom = true;
86             }
87     }
88 }
89
90 public final void deallocate() { ... }
91
92 private void clearParamsLetters() { ... }
93 }

```

```

1 public class GameDragAndDropController {
2
3     private DragAndDropGameView dragAndDropGameView;
4
5     private ImageViewAnimationTTS itemImage;
6
7     public OnNextItemListener onNextItemListener;
8
9     private VibrationFeedback vibrationFeedback;
10
11     private SimpleSoundPool sfx;
12
13     private Random random;
14
15     private Resources resources;
16
17     private SparseArrayCompat<DragItem> dragItems;
18
19     private static int itemIndex = 0;
20
21     private Handler nextItemHandler;
22
23     public TextToSpeech textToSpeech;
24
25     public GameDragAndDropController(@NonNull Context context,
26                                     @NonNull final View rootViewLayout,
27                                     @NonNull final LayoutInflater inflater) {
28
29         resources = context.getResources();
30
31         ( ... )
32
33         if (random == null) {
34             random = new Random();
35         }
36
37         dragAndDropGameView = rootViewLayout.findViewById(R.id.
38             word_view_layout_container);
39
40         itemImage = rootViewLayout.findViewById(R.id.item_image);
41
42

```

```

43
44     if (dragItems == null) {
45         dragItems = new SparseArrayCompat<DragItem>(ItemStruct.ITEM_SIZE);
46     }
47
48     startItemLetter();
49 }
50
51 public final void playFeedbackButton() { ... }
52
53 public final void deallocate() { ... }
54
55 private void startItemLetter() { ... }
56
57 private void startLevel() { ... }
58
59 public final void nextItemLetter() { ... }
60
61 private void createItemLetters() { ... }
62
63 private void stopTTS() { ... }
64
65 public final void setTTS() { ... }
66
67 }

```

4.3.9 Módulo – Framework

Foi desenvolvido um módulo que tem o comportamento de uma library, contém todo o código-fonte reutilizável por qualquer um dos módulos que compõe a aplicação final ou qualquer projeto que utilize esta library como dependência. O objetivo é centralizar as funcionalidades comuns e evitar códigos repetidos.

Este módulo contém diversas funcionalidades, como manipulação de áudio e vídeo, componentes customizados e adaptados para facilitar a reutilização (imagens, textos, caixas de diálogos, temas, etc.), persistência de dados, processos assíncronos e a conversão de texto para voz. A seguir serão demonstrados exemplos das funcionalidades implementadas neste módulo:

GetTextRawFile: Esta classe tem a funcionalidade de buscar um ficheiro .txt localizado na pasta raw de um projeto Android e retornar um objeto da classe `CharSequence`¹⁵ com o conteúdo deste ficheiro. Esta funcionalidade é útil para inserir uma grande quantidade de texto num componente do tipo `TextView`¹⁶.

¹⁵ **Classe `CharSequence`:** <https://developer.android.com/reference/java/lang/CharSequence>, acedido em 25/04/2019

¹⁶ **Classe `TextView`:** <https://developer.android.com/reference/android/widget/TextView>, acedido em 25/04/2019


```

1 public class GetTextRawFile {
2
3     public static CharSequence getTextRawFile(@NonNull final Resources res,
4                                               final int resId) {
5
6         CharSequence charSequence = null;
7
8         try {
9             final InputStream raw = res.openRawResource(resId);
10
11             if (raw != null && raw.available() > 0) {
12
13                 final byte[] reader = new byte[raw.available()];
14
15                 while (raw.read(reader) != -1) {
16                     // lentidao ao ler ficheiros no raw
17                 }
18
19                 if (reader.length > 0) {
20                     charSequence = new String(reader);
21                 }
22                 raw.close();
23             }
24         } catch (IOException e) {
25             e.printStackTrace();
26         }
27         return charSequence;
28     }
29 }

1 // Exemplo para carregar todas as paginas do livro em ficheiros separados
2 private final void createPageText(@NonNull final Resources res) {
3     if (pageText == null) {
4         final int pageLength = BookResources.BOOK_PAGES_TEXT.length;
5         pageText = new SparseArrayCompat<CharSequence>(pageLength);
6         final int size = BookResources.BOOK_PAGES_TEXT.length;
7
8         for (int i = 0; i < size; i++) {
9             final int pageIndexID = BookResources.BOOK_PAGES_TEXT[i];
10             final CharSequence page = GetTextRawFile.getTextRawFile(res, pageIndexID);
11
12             if (page != null) {
13                 pageText.put(i, page);
14             }
15         }
16     }
17 }

```

PersistenceManager: Esta classe tem a funcionalidade de facilitar a persistência e recuperação de alguma informação simples, que não necessite de uma base de dados, como por exemplo guardar um inteiro que represente a página atual do livro.

```
1 public class PersistenceManager {
2
3     public static void saveSharedPreferencesString(@NonNull Context context,
4                                                     final String key,
5                                                     final String value) {
6         if (key != null && value != null) {
7             final SharedPreferences sharedPrefs =
8                 PreferenceManager.getDefaultSharedPreferences(context);
9
10            if (sharedPrefs != null) {
11                final SharedPreferences.Editor editor = sharedPrefs.edit();
12                if (editor != null) {
13                    editor.putString(key, value);
14                    editor.commit();
15                }
16            }
17        }
18    }
19
20    public static String loadSharedPreferencesString(@NonNull Context context,
21                                                     final String key,
22                                                     final String defaultValue) {
23        String value = defaultValue;
24
25        if (key != null && defaultValue != null) {
26            final SharedPreferences sharedPrefs =
27                PreferenceManager.getDefaultSharedPreferences(context);
28
29            if (sharedPrefs != null) {
30                value = sharedPrefs.getString(key, defaultValue);
31            }
32        }
33        return value;
34    }
35
36    public static void saveSharedPreferencesInt(@NonNull Context context,
37                                                final String key,
38                                                final int value) { ... }
39
40    public static int loadSharedPreferencesInt(@NonNull Context context,
41                                                final String key,
42                                                final int defaultValue) { ... }
43
44    public static void saveSharedPreferencesBoolean(@NonNull Context context,
45                                                    final String key,
46                                                    final boolean value) { ... }
47
48    public static boolean loadSharedPreferencesBoolean(@NonNull Context context,
49                                                       final String key,
50                                                       final boolean defaultValue)
51        { ... }
52
53 }
```

```

1 // Exemplo da utilizacao da funcionalidade da classe PersistenceManager
2 public final void savePage() {
3
4     if (EduardoStuff.ENABLE_SAVE_PAGE_BOOK) {
5         PersistenceManager.saveSharedPreferencesInt(context,
6                                                     PERSISTENCE_KEY_PAGE,
7                                                     pageNumber);
8     }
9 }
10
11 public final void loadPage() {
12
13     if (EduardoStuff.ENABLE_SAVE_PAGE_BOOK) {
14         pageNumber = PersistenceManager.loadSharedPreferencesInt(context,
15                                                                 PERSISTENCE_KEY_PAGE,
16                                                                 0);
17     }
18     goBookPage(pageNumber);
19 }

```

MediaPlayerManager: Esta classe é utilizada para simplificar o uso de player de áudio.

```

1 public class MediaPlayerManager {
2     private MediaPlayer mediaPlayer;
3     private final Context context;
4     private Uri mediaUri;
5     private int currentPosition = 0;
6
7     public MediaPlayerManager(final String mediaPath, final Context context) {
8         this.context = context;
9         mediaUri = Uri.parse(mediaPath);
10    }
11
12    public final void resetMedia(final boolean isLooping) {
13        if (mediaPlayer != null) {
14            mediaPlayer.reset();
15            try {
16                mediaPlayer.setDataSource(context, mediaUri);
17                mediaPlayer.prepareAsync();
18            } catch (Exception e) {
19                e.printStackTrace();
20            }
21            if (mediaUri != null) {
22                try {
23                    mediaPlayer.setLooping(isLooping);
24                    mediaPlayer
25                        .setOnPreparedListener(new MediaPlayer.OnPreparedListener()
26                        {
27                            @Override
28                            public void onPrepared(MediaPlayer arg0) {
29                                mediaPlayer.start();
30                            }
31                        });
32                } catch (Exception e) {
33                    e.printStackTrace();
34                }
35            }
36        }
37    }
38 }

```

```

37
38     public final void playMedia(final boolean isLooping, final boolean start) {
39         if (mediaUri != null) {
40             if (mediaPlayer == null) {
41                 mediaPlayer = new MediaPlayer();
42                 mediaPlayer.setAudioStreamType(AudioManager.STREAM_MUSIC);
43                 try {
44                     mediaPlayer.setDataSource(context, mediaUri);
45                     mediaPlayer.prepareAsync();
46                 } catch (Exception e) {
47                     e.printStackTrace();
48                 }
49             }
50             if (mediaUri != null) {
51                 try {
52                     mediaPlayer.setLooping(isLooping);
53
54                     mediaPlayer
55                         .setOnPreparedListener(new MediaPlayer.OnPreparedListener()
56                             {
57                                 @Override
58                                 public void onPrepared(MediaPlayer arg0) {
59                                     if (start) {
60                                         mediaPlayer.start();
61                                     }
62                                 }
63                             });
64                 } catch (Exception e) {
65                     e.printStackTrace();
66                 }
67             }
68         }
69
70     public final void pauseMedia() { ... }
71
72     public final void resumeMedia() { ... }
73
74     public final void stopMedia() { ... }
75
76     public final void releaseMedia() { ... }
77
78     public final boolean isPlay() { ... }
79 }

```

```

1 // Exemplo do uso da classe MediaPlayerManager
2 // musicPath: caminho do ficheiro do audio localizada geralmente na pasta
3 // do projeto Android ou URI numa pasta externa
4 if (musicPath != null && musicPath.length() > 0) {
5     music = new MediaPlayerManager(musicPath, context);
6     if (music != null) {
7         if (GlobalSettings.soundEnable) {
8             music.playMedia(false, GlobalSettings.soundEnable);
9         } else {
10             music.pauseMedia();
11         }
12     }
13 }

```

SimpleSoundPool: Esta classe é utilizada para simplificar o uso de efeitos sonoros.

```
1 public class SimpleSoundPool {
2
3     private static final int MAX_QUALITY = 100;
4     private static final float NORMAL_SPEED = 1.0f;
5     private static final float VOLUME = 1.0f;
6     private static final int VOLUME_NONE = -1;
7
8     private SoundPool soundPool;
9
10    private SparseIntArray sfxMapInt;
11
12    private boolean loaded = true;
13
14    private AudioManager audioManager;
15
16    public SimpleSoundPool(@NonNull final Context context, final int... sfxResIDArray) {
17
18        final int sizeSfx = sfxResIDArray.length;
19
20        if (sizeSfx > 0) {
21
22            if (audioManager == null) {
23                audioManager = (AudioManager) context.getSystemService(Context.
24                    AUDIO_SERVICE);
25            }
26
27            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
28                createNewSoundPool(sizeSfx);
29            } else {
30                createOldSoundPool(sizeSfx);
31            }
32
33            if (sfxMapInt == null) {
34                sfxMapInt = new SparseIntArray(sizeSfx);
35            }
36
37            soundPool.setOnLoadCompleteListener(new OnLoadCompleteListener() {
38                @Override
39                public void onLoadComplete(SoundPool soundPool, int sampleId,
40                    int status) {
41                    loaded = true;
42                }
43            });
44
45            /**
46             * SoundPool retorna um id apos realizar o loading do resource,
47             * sendo que este id eh armazenado num mapping para buscar este valor
48             * de acordo com o id do resources
49             */
50            for (int sfxResId : sfxResIDArray) {
51                final int indexSoundPool = soundPool.load(context, sfxResId, MAX_QUALITY
52                    );
53                sfxMapInt.put(sfxResId, indexSoundPool);
54            }
55        }
56    }
```

```

57     public void playSound(final int sfx) {
58
59         final boolean check = (soundPool != null && audioManager != null &&
60             loaded && sfxMapInt != null && sfxMapInt.size() > 0);
61
62         if (check) {
63             float streamVolume = VOLUME_NONE;
64
65             streamVolume = audioManager.getStreamVolume(AudioManager.STREAM_MUSIC);
66
67             final float actualVolume = (float) audioManager.getStreamVolume(AudioManager
68                 .STREAM_MUSIC);
69
70             final float maxVolume = (float) audioManager.getStreamMaxVolume(AudioManager
71                 .STREAM_MUSIC);
72
73             streamVolume = actualVolume / maxVolume;
74
75             final float volume = (streamVolume == VOLUME_NONE) ? VOLUME : streamVolume;
76
77             soundPool.play(sfxMapInt.get(sfx), volume, volume, 1, 0, NORMAL_SPEED);
78         }
79     }
80
81     private void createNewSoundPool(final int sizeSfx) {
82
83         if (soundPool == null) {
84             final AudioAttributes attributes = new AudioAttributes.Builder()
85                 .setUsage(AudioAttributes.USAGE_GAME)
86                 .setContentType(AudioAttributes.CONTENT_TYPE_SONIFICATION)
87                 .build();
88
89             if (attributes != null) {
90                 soundPool = new SoundPool.Builder().setAudioAttributes(
91                     attributes).build();
92             }
93         }
94     }
95
96     private void createOldSoundPool(final int sizeSfx) {
97         if (soundPool == null) {
98             soundPool = new SoundPool(sizeSfx, AudioManager.STREAM_MUSIC, MAX_QUALITY);
99         }
100     }
101
102     public void release() { ... }
103 }

```



```

1 // Exemplo do uso que criar um pool com 3 efeitos sonoros
2 // Para tocar o som usa-se o id como referencia
3 SimpleSoundPool sfx = new SimpleSoundPool(context, R.raw.sfx_normal_click,
4                                     R.raw.sfx_winner,
5                                     R.raw.sfx_lose);
6 sfx.playSound(R.raw.sfx_normal_click);
7 sfx.playSound(R.raw.sfx_winner);

```

VibrationFeedback: Esta classe é utilizada no feedback por vibração no dispositivo.

```
1 public class VibrationFeedback {
2
3     private Vibrator vibrator;
4
5     private static final long VIBRATE_MILLISECONDS_DEFAULT = 100;
6
7     public VibrationFeedback(@NonNull final Context context) {
8
9         if (vibrator == null) {
10
11             if (checkVibratePermission(context)) {
12                 vibrator = (Vibrator) context.
13                     getSystemService(Context.VIBRATOR_SERVICE);
14             }
15
16         }
17     }
18
19     private boolean checkVibratePermission(@NonNull final Context context) {
20
21         boolean result = false;
22
23         final String permission = "android.permission.VIBRATE";
24
25         final int res = context.checkCallingOrSelfPermission(permission);
26
27         result = (res == PackageManager.PERMISSION_GRANTED);
28
29         return result;
30     }
31
32     public final void vibrate(final long milliseconds) {
33
34         if (vibrator != null && milliseconds > 0) {
35             vibrator.vibrate(milliseconds);
36         }
37     }
38
39     public final void vibrate() {
40
41         if (vibrator != null) {
42             vibrator.vibrate(VIBRATE_MILLISECONDS_DEFAULT);
43         }
44     }
45 }

```



```
1 // Exemplo do uso da classe VibrationFeedback
2 if (vibrationFeedback == null) {
3     vibrationFeedback = new VibrationFeedback(getActivity());
4 }
5 vibrationFeedback.vibrate();

```

ReducedCustomPagerAdapter: Esta classe é utilizada para criar um PagerAdapter¹⁷.

```
1 public class ReducedCustomPagerAdapter extends PagerAdapter {
2     private LayoutInflater inflater;
3     private SparseArrayCompat<View> listViews;
4
5     public ReducedCustomPagerAdapter(@NonNull LayoutInflater inflater) {
6         this.inflater = inflater;
7     }
8
9     public final void addViewById(final int idView, final int position) {
10        if (idView > 0) {
11            if (listViews == null) {
12                listViews = new SparseArrayCompat<View>();
13            }
14            final View view = inflater.inflate(idView, null);
15
16            if (view != null) {
17                listViews.put(position, view);
18            }
19        }
20    }
21
22    @Override
23    public final boolean isViewFromObject(final View view, final Object object) {
24
25        if (view != null && object != null) {
26            return view == object;
27        }
28        return false;
29    }
30
31    @Override
32    public final void destroyItem(final View view, final int index,
33                                final Object object) {
34
35        if (view != null && index >= 0 && object != null) {
36            ((ViewPager) view).removeView((View) object);
37        }
38    }
39
40    @Override
41    public final Object instantiateItem(final View view, final int position) {
42        View viewItem = null;
43        if (view != null && position >= 0) {
44            viewItem = listViews.get(position);
45            view.addView(viewItem);
46        }
47        return viewItem;
48    }
49
50    public final View getViewByIndex(final int index) {
51        View view = null;
52        if (listViews != null) {
53
54            final int size = listViews.size();
55
```

¹⁷ Classe PagerAdapter: <https://developer.android.com/reference/android/support/v4/view/PagerAdapter>, acessado em 25/04/2019


```

56         if (index >= 0 && index < size) {
57             view = listViews.get(index);
58         }
59     }
60     return view;
61 }
62
63 public final void deallocate() {
64     if (listViews != null) {
65         final int size = listViews.size();
66
67         if (size > 0) {
68             listViews.clear();
69             listViews = null;
70         }
71     }
72 }
73 }

1 // Exemplo do uso da classe ReducedCustomPagerAdapter
2 private ReducedCustomPagerAdapter adapter;
3
4 private void createAdapter (final LayoutInflater inflater) {
5     if (context != null && adapter == null && inflater != null) {
6         adapter = new ReducedCustomPagerAdapter(inflater);
7         final int size = BookSPCResources.LAYOUTS_LENGTH;
8         for (int i = 0; i < size; i++) {
9             adapter.addViewById(BookSPCResources.BOOK_PAGES_LAYOUT[i], i);
10        }
11    }
12 }
13
14 private void createViewPager () {
15     if (adapter != null) {
16         viewPager.setAdapter(adapter);
17         viewPager.setOnPageChangeListener(new OnPageChangeListener() {
18
19             @Override
20             public void onPageSelected (final int page) {
21                 pageNumber = page;
22                 pageNumberText = "" + (pageNumber + 1);
23
24                 if (menuItemPageNumber != null) {
25                     menuItemPageNumber.setTitle(pageNumberText);
26                 }
27             }
28
29             @Override
30             public void onPageScrolled (final int page,
31                 final float positionOffset,
32                 final int positionOffsetPixels) {
33             }
34
35             @Override
36             public void onPageScrollStateChanged (final int position) {
37             }
38         });
39     }
40 }

```

LoadTask: Esta classe é utilizada para facilitar o uso da classe AsyncTask, implementando 2 interfaces chamadas OnLoadTaskExecute e OnLoadTaskFinish.

```
1 public class LoadTask extends AsyncTask<Void, Void, Void> {
2
3     private final OnLoadTaskFinish onLoadTaskFinish;
4     private final OnLoadTaskExecute onLoadTaskExecute;
5
6     public LoadTask(final OnLoadTaskFinish onLoadTaskFinish,
7                     final OnLoadTaskExecute onLoadTaskExecute) {
8         this.onLoadTaskFinish = onLoadTaskFinish;
9         this.onLoadTaskExecute = onLoadTaskExecute;
10    }
11
12    @Override
13    protected void onPreExecute() { ... }
14
15    @Override
16    protected Void doInBackground(final Void... params) {
17        if (onLoadTaskExecute != null) {
18            onLoadTaskExecute.onLoadTaskExecute();
19        }
20    }
21
22    @Override
23    protected void onPostExecute(final Void result) {
24        if (onLoadTaskFinish != null) {
25            onLoadTaskFinish.onLoadTaskFinish();
26        }
27    }
28 }

```



```
1 public interface OnLoadTaskExecute {
2
3     public void onLoadTaskExecute();
4 }

```



```
1 public interface OnLoadTaskFinish {
2
3     public void onLoadTaskFinish();
4 }

```



```
1 // Exemplo do uso da classe LoadTask
2 OnLoadTaskExecute onLoadTaskExecute = new OnLoadTaskExecute() {
3     @Override
4     public void onLoadTaskExecute() {
5         loadImages();
6     }
7 };
8
9 OnLoadTaskFinish onLoadTaskFinish = new OnLoadTaskFinish() {
10    @Override
11    public void onLoadTaskFinish() {
12        final Intent mainActivityIntent = new Intent(SplashScreenActivity.this, cls);
13        SplashScreenActivity.this.startActivity(mainActivityIntent);
14    }
15 };
16
17 LoadTask loadTask = new LoadTask(this, this);
18 loadTask.execute();

```

UIComponentUtils: Esta classe é utilizada para manipulação de imagens.

```
1 public class UIComponentUtils {
2
3     private static final Bitmap BITMAP_EMPTY = null;
4
5     public static StateListDrawable createStateListDrawable(final Theme theme,
6                                                             final int attrOn,
7                                                             final int attrOff,
8                                                             final int gravity) { ... }
9
10    public static Drawable getDrawableById(final Theme theme,
11                                           final int res_image_id) { ... }
12
13    public static Bitmap getBitmapById(final Context context,
14                                       final int image_res_id) { ... }
15
16    public static int getDimensionPixelSizeByTheme(final Theme theme,
17                                                  final int attrValue,
18                                                  final int defaultValue) { ... }
19
20    public static Bitmap resizeBitmapByDrawableId(final Context context,
21                                                  final int new_width,
22                                                  final int new_height,
23                                                  final int resIdTheme) { ... }
24
25    public static Bitmap resizeBitmapById(final Context context,
26                                         final int new_width,
27                                         final int new_height,
28                                         int image_res_id) { ... }
29
30    public static Bitmap resizeBitmapById(final Context context,
31                                         final int new_height,
32                                         int image_res_id) { ... }
33
34    public static Bitmap decodeSampledBitmapFromResource(Resources res,
35                                                         int resId, int reqWidth,
36                                                         int reqHeight) { ... }
37
38    private static int calculateInSampleSize(BitmapFactory.Options options,
39                                             int reqWidth, int reqHeight) { ... }
40 }

1 // Exemplo do uso da classe UIComponentUtils
2 private final void createPageImageLow(@NonNull final Context context) {
3
4     final int size = BookResources.BOOK_PAGES_IMAGE.length;
5
6     final int height = DeviceDimensionsUtil.getDisplayHeight(context) / 2;
7
8     final int width = height + (height / 4);
9
10    if (pageImages == null) {
11        pageImages = new SparseArrayCompat<Drawable>(size);
12
13        final Resources resources = context.getResources();
14
15        if (resources != null) {
16
17            BitmapDrawable bkg_image_temp = null;
```

```

18
19         for (int i = 0; i < size; i++) {
20             final int pageIndexID = BookResources.BOOK_PAGES_IMAGE[i];
21
22             if (pageIndexID != -1) {
23                 bkg_image_temp = new BitmapDrawable(resources,
24                     UIComponentUtils.decodeSampledBitmapFromResource(resources,
25                                                                 pageIndexID,
26                                                                 width,
27                                                                 height));
28
29                 pageImages.put(i, bkg_image_temp);
30             } else {
31                 pageImages.put(i, null);
32             }
33         }
34     }
35 }
36 }

```

ImageViewAnimationTTS: Esta classe é uma adaptação da classe `ImageView`¹⁸ com as funcionalidades para efetuar uma animação e converter o texto para voz.

```

1  public class ImageViewAnimationTTS extends ImageView implements OnClickListener {
2
3      private Animation animation;
4      private OnClickListener clickListener;
5      private String text;
6      private TextToSpeech textToSpeech;
7
8      public ImageViewAnimationTTS(final Context context) { ... }
9
10     @Override
11     public final void setOnClickListener(final OnClickListener onClickListener) {
12         if (onClickListener == this) {
13             super.setOnClickListener(onClickListener);
14         } else {
15             clickListener = onClickListener;
16         }
17     }
18
19     @Override
20     public final void onClick(final View view) {
21         if (view != null) {
22             if (animation != null) {
23                 view.startAnimation(animation);
24             }
25
26             if (textToSpeech != null) {
27                 if (text != null && text.length() > 0) {
28                     if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
29                         speakNewApiTTS();
30                     } else {
31                         speakOldApiTTS();
32                     }
33                 }
34             }
35         }
36     }
37 }

```

¹⁸ **Classe `ImageView`:** <https://developer.android.com/reference/android/widget/ImageView>, acedido em 25/04/2019

```

35         if (clickListener != null) {
36             clickListener.onClick(this);
37         }
38     }
39 }
40
41 private void createAnimation(final AttributeSet attrs) {
42     if (animation == null) {
43
44         final TypedArray typedArray = getContext()
45             .obtainStyledAttributes(attrs,
46                 R.styleable.imageview_animation);
47
48         if (typedArray != null) {
49             final int reference = typedArray.getResourceId(R.styleable.
50                 imageview_animation_layout_animation, 0);
51
52             if (reference != 0) {
53                 animation = AnimationUtils.loadAnimation(getContext(), reference);
54             }
55
56             text = typedArray.getString(R.styleable.
57                 imageview_animation_text_animation);
58
59             typedArray.recycle();
60         }
61     }
62
63     public void setTextToSpeech(final TextToSpeech textToSpeech) { ... }
64
65     public final void setText(final String textTTS) { ... }
66
67     private void speakOldApiTTS() { ... }
68
69     private void speakNewApiTTS() { ... }
70 }

```



```

1 <com.toymobi.framework.view.imageview.ImageViewAnimationTTS
2     android:id="@+id/imageViewSPC_1_1_1"
3     android:layout_width="wrap_content"
4     android:layout_height="wrap_content"
5     android:padding="@dimen/button_item_spc_padding"
6     android:src="@drawable/spc_eduardo"
7     imageview_animation:layout="@anim/rotate_anti_clock_wise"
8     imageview_animation:text="@string/spc_1_1_1" />

```

LockableHorizontalScrollView: Esta classe foi implementada para corrigir um problema na rolagem vertical e horizontal com o componente de swipe e scroll na mesma página. A correção foi manter o evento preso até terminar o movimento de scroll na horizontal.

```

1  public class LockableHorizontalScrollView extends HorizontalScrollView {
2
3      public LockableHorizontalScrollView (final Context context) { ... }
4
5      @Override
6      public boolean onTouchEvent (final MotionEvent motionEvent) {
7
8          // correcao no scroll na vertical e horizontal
9          final int action = motionEvent.getAction();
10
11         if (action == MotionEvent.ACTION_MOVE && getParent() != null) {
12             getParent().requestDisallowInterceptTouchEvent(true);
13         }
14         return super.onTouchEvent(motionEvent);
15     }
16
17     @Override
18     public boolean onInterceptTouchEvent (MotionEvent motionEvent) {
19
20         if (getParent() != null) {
21
22             // quando estiver movendo na horizontal bloqueia o scroll na vertical
23             switch (motionEvent.getAction()) {
24
25                 case MotionEvent.ACTION_MOVE:
26                     getParent().requestDisallowInterceptTouchEvent(true);
27                     break;
28
29                 case MotionEvent.ACTION_UP:
30                 case MotionEvent.ACTION_CANCEL:
31                     getParent().requestDisallowInterceptTouchEvent(false);
32                     break;
33             }
34         }
35         return super.onInterceptTouchEvent(motionEvent);
36     }
37 }

```

```

1  <com.toymobi.framework.view.scrollview.LockableHorizontalScrollView
2      android:layout_width="match_parent"
3      android:layout_height="match_parent"
4      android:background="@null"
5      android:clickable="true"
6      android:fadingEdge="none"
7      android:scrollbars="none" >
8
9
10     <LinearLayout
11         android:layout_width="wrap_content"
12         android:layout_height="wrap_content"
13         android:gravity="center_vertical"
14         android:orientation="horizontal" >
15

```

```

16
17         <com.toymobi.framework.view.imageview.ImageViewAnimationTTS
18             android:id="@+id/imageViewSPC_1_1_1"
19             android:layout_width="wrap_content"
20             android:layout_height="wrap_content"
21             android:padding="@dimen/button_item_spc_padding"
22             android:src="@drawable/spc_eduardo"
23             imageview_animation:layout="@anim/rotate_anti_clock_wise"
24             imageview_animation:text="@string/spc_1_1_1" />
25
26         ...
27
28     </LinearLayout>
29
30 </com.toymobi.framework.view.scrollview.LockableHorizontalScrollView>
31
32
33
34 <com.toymobi.framework.view.textview.TextViewCustomFont
35     style="@style/TextBook"
36     android:text="@string/text_1_1"
37     android:textColor="@color/ACTION_BAR_COLOR"
38     custom_text_view_font:font="@string/roboto_black" />

```

4.3.10 Módulo – Recursos

Contém um repositório de recursos audiovisuais (texto, imagem e áudio) reutilizáveis pelos principais módulos (aplicação principal, livros multimédias e jogos), com o objetivo de centralizar tudo que for comum, evitando replicação desnecessária dos recursos.

Neste módulo estão todas as ilustrações do livro, sendo utilizadas em 4 micro-services (livro básico, livro spc, livro lgp e o jogo quebra-cabeça), além dos ficheiros com efeitos sonoros que são os mesmos para todos os módulos que compõe a aplicação final.

Contém valores e referências comuns para todos os módulos, centralizando opções como dimensões, cores, etc., conforme demonstram alguns exemplos a seguir:

```

1  <resources>
2      <color name="SNOW">#FFFAFA</color>
3      <color name="DARK_GRAY_SHADOWN">#22000000</color>
4      <color name="ACTION_BAR_COLOR">#00579B</color>
5      <color name="BACKGROUND_COLOR">#008FD4</color>
6      <color name="DARK_BLUE">#0099CC</color>
7      <color name="RED">#E51C23</color>
8      <color name="PINK">#E91E63</color>
9      <color name="PURPLE">#9C27B0</color>
10     <color name="INDIGO">#1A237E</color>
11     <color name="LIGHT_BLUE">#03A9F4</color>
12     <color name="GREEN">#259B24</color>
13     <color name="YELLOW">#FFD600</color>
14     <color name="ORANGE">#FF5722</color>
15     <color name="BROWN">#795548</color>
16     <color name="BLACK">#000000</color>
17     <color name="WHITE">#FFFFFF</color>
18 </resources>

1  <resources>
2      <!-- Text Default -->
3      <dimen name="text_max">16sp</dimen>
4      <dimen name="text_med">14sp</dimen>
5      <dimen name="text_min">12sp</dimen>
6
7      <!-- Text Size Dialogs -->
8      <dimen name="dialog_padding_max">16dip</dimen>
9      <dimen name="dialog_padding_med">8dip</dimen>
10     <dimen name="dialog_padding_min">4dip</dimen>
11     <dimen name="dialog_padding_fixed_ok">6dip</dimen>
12
13     <!-- padding/margin Scroll Size Dialogs -->
14     <dimen name="scroll_margin_left">8dp</dimen>
15     <dimen name="scroll_margin_right">6dp</dimen>
16 </resources>

1  <resources xmlns:android="http://schemas.android.com/apk/res/android">
2      <style name="TextTitleActionBar">
3          <item name="android:background">@android:color/transparent</item>
4          <item name="android:textStyle">bold</item>
5          <item name="android:layout_width">wrap_content</item>
6          <item name="android:layout_height">match_parent</item>
7          <item name="android:clickable">false</item>
8          <item name="android:textColor">@color/SNOW</item>
9          <item name="android:shadowColor">@color/DARK_GRAY_SHADOWN</item>
10         <item name="android:shadowDx">4</item>
11         <item name="android:shadowDy">4</item>
12         <item name="android:shadowRadius">1</item>
13     </style>
14
15     <style name="StyleTextMedLight">
16         <item name="android:background">@android:color/transparent</item>
17         <item name="android:textStyle">bold</item>
18         <item name="android:layout_width">match_parent</item>
19         <item name="android:layout_height">wrap_content</item>
20         <item name="android:textSize">@dimen/text_med</item>
21         <item name="android:gravity">center</item>
22         <item name="android:singleLine">true</item>
23         <item name="android:shadowColor">@color/DARK_GRAY_SHADOWN</item>
24         <item name="android:shadowDx">4</item>

```



```

25         <item name="android:shadowDy">4</item>
26         <item name="android:shadowRadius">1</item>
27     </style>
28
29     <style name="StyleTextBigLight">
30         <item name="android:background">@android:color/transparent</item>
31         <item name="android:textStyle">bold</item>
32         <item name="android:layout_width">match_parent</item>
33         <item name="android:layout_height">wrap_content</item>
34         <item name="android:textSize">@dimen/text_max</item>
35         <item name="android:gravity">center</item>
36         <item name="android:singleLine">true</item>
37         <item name="android:shadowColor">@color/DARK_GRAY_SHADOWN</item>
38         <item name="android:shadowDx">4</item>
39         <item name="android:shadowDy">4</item>
40         <item name="android:shadowRadius">1</item>
41     </style>
42
43     <style name="StyleTextLongInfo">
44         <item name="android:background">@android:color/transparent</item>
45         <item name="android:textStyle">bold</item>
46         <item name="android:layout_width">match_parent</item>
47         <item name="android:layout_height">wrap_content</item>
48         <item name="android:textSize">@dimen/text_max</item>
49         <item name="android:gravity">center</item>
50         <item name="android:shadowColor">@color/DARK_GRAY_SHADOWN</item>
51         <item name="android:shadowDx">4</item>
52         <item name="android:shadowDy">4</item>
53         <item name="android:shadowRadius">1</item>
54     </style>
55 </resources>

```

```

1 <resources>
2     <style name="Theme.NoBackground" parent="android:Theme">
3         <item name="android:windowBackground">@null</item>
4         <item name="android:windowNoTitle">true</item>
5         <item name="android:windowFullscreen">true</item>
6         <item name="android:soundEffectsEnabled">false</item>
7     </style>
8
9     <!-- the theme applied to the application or activity -->
10    <style name="CustomActionBarTheme" parent="@style/Theme.AppCompat.Light.
        DarkActionBar">
11
12        <!-- Support library compatibility -->
13        <item name="actionBarStyle">@style/MyActionBar</item>
14        <item name="android:windowFullscreen">true</item>
15        <item name="android:soundEffectsEnabled">false</item>
16    </style>
17
18    <!-- ActionBar styles -->
19    <style name="MyActionBar" parent="@style/Widget.AppCompat.Light.ActionBar.Solid.
        Inverse">
20        <item name="android:background">@color/ACTION_BAR_COLOR</item>
21        <!-- Support library compatibility -->
22        <item name="background">@color/ACTION_BAR_COLOR</item>
23        <item name="android:windowFullscreen">true</item>
24    </style>
25 </resources>

```

Foi implementada uma classe com parâmetros e funcionalidades comuns para todos os módulos, como chaves para persistência e caixas de diálogos, como demonstra o exemplo do código-fonte a seguir:

```
1 public class EduardoStuff {
2
3     public static boolean isMainMenu = true;
4
5     public static final String PERSISTENCE_KEY = "MACARENA";
6
7     public static boolean ENABLE_TTS = true;
8
9     public static boolean ENABLE_SAVE_PAGE_BOOK = false;
10
11     public static Random random = new Random();
12
13     public static final void showMessageExitSPC(final Context context) { ... }
14
15     public static final void showMessageBackMainMenuSPC(final Context context) { ... }
16
17     private static final void exit(final Context context) { ... }
18
19     private static final void goHome(final Context context) { ... }
20
21
22     public static final void showCongratsDialog(final Context context,
23                                                 final CharSequence infoText,
24                                                 final OnClickListener ok) { ... }
25
26     public static final void setLanguage(final Context context) { ... }
27
28     public static final boolean isNormalSize(final Context context) { ... }
29 }
```

Capítulo 5

Testes e Resultados

Neste capítulo são demonstrados os testes e os resultados obtidos da aplicação nas lojas digitais. Apresentando a relação deste projeto com o empreendedorismo através da participação em programas de aceleração e concursos de ideias. Mostra os resultados e as divulgações que ocorreram nos meios de comunicação. Serão apresentadas ideias que surgiram durante o projeto, estando relacionadas com as áreas da educação, terapias e reabilitação.

5.1 Testes

Foram realizados vários testes utilizando diversos dispositivos da plataforma Android, os modelos foram divididos em três categorias de acordo com as principais características (processamento, memória e resolução do ecrã), conforme demonstrado a seguir:

- **Baixa gama:** Sony Xperia U, Tablet Deco Proteste, Samsung Galaxy Core Prime, Samsung Galaxy J3 e Tablet Magalhães Mg5t .
- **Média gama:** Google Nexus 7, Samsung Galaxy Tab 10.1, Huawei P9 e Motorola Moto G4 Plus.
- **Alta Gama:** Motorola Moto Maxx, Samsung Galaxy S5, Samsung Galaxy Tab 3 e Samsung Note 4.

Os principais casos de testes foram elaborados e validados de acordo com as diretrizes e critérios de qualidade definidos pela plataforma Android, disponível em “Qualidade da aplicação principal” ¹. Os princípios contidos na tabela 5.1 foram considerados durante o processo de desenvolvimento desta aplicação.

¹ **Qualidade da aplicação principal:** <https://developer.android.com/docs/quality-guidelines/core-app-quality>, acessado em 20/12/2018

Tabela 5.1: Principais casos de testes

CT01	Não redefinir a funcionalidade do botão “Voltar” e “Home” com funções diferentes do padrão.
CT02	Todas as caixas de dialogo devem fechar com a funcionalidade do botão “Voltar”.
CT03	Solicitar apenas as permissões necessárias para as funcionalidades da aplicação.
CT04	Não solicitar permissões para aceder a dados privados do utilizador (contatos e os históricos do sistema) ou serviços que podem gerar custos (ligar e mensagens de SMS).
CT05	A aplicação deve funcionar normalmente quando for instalada no armazenamento externo (cartão de memória do dispositivo).
CT06	O áudio não deve ser reproduzido quando a aplicação está em segundo plano.
CT07	O áudio deve ser reiniciado quando a aplicação volta para o primeiro plano.
CT08	A aplicação não deve deixar serviços em execução quando estiver em segundo plano.
CT09	A aplicação tem que preservar e restaurar o estado do utilizador quando for pausada e resumida.
CT010	A aplicação não deve conter nenhum problema que impacte no encerramento ou bloqueio da aplicação.
CT011	A aplicação deve iniciar rapidamente ou fornecer algum indicador para o utilizador que o carregamento pode levar mais tempo.
CT012	A aplicação deve reproduzir áudio e vídeos sem problemas.
CT013	A aplicação fornece gráficos com qualidade para todos os tamanhos de ecrã.
CT014	A aplicação exhibe textos de forma aceitável para uma fácil leitura do utilizador.

Foram realizadas várias apresentações com os utilizadores em diversas instituições e em eventos relacionados com a educação especial e pessoas com deficiência. Alguns locais foram o Agrupamento de Escolas Afonso de Albuquerque, Jardim de Infância de Maçainhas, ADM Estrela, Fundação Augusto Gil, Aldeia de Crianças SOS, ABPG – Associação de Beneficência Popular de Gouveia, Escola Básica Professor José Salvado Sampaio e o Agrupamento de Escolas Quinta de Marrocos ambos em Lisboa. As figuras 5.1 e 5.2 demonstram algumas das apresentações realizadas.



Figura 5.1: Apresentações com os utilizadores I



Figura 5.2: Apresentações com os utilizadores II

Em relação aos testes de usabilidade, será realizado como um trabalho futuro. Durante o desenvolvimento deste projeto não foi possível colocar em prática as metodologias para a execução e coleta de resultados. Como por exemplo usar a Avaliação Heurística, definida por Nielsen e Molich² em 1994.

Um dos principais desafios para realizar o teste de usabilidade foi o perfil dos participantes, na maior parte eram crianças na fase de pré-alfabetização e pessoas (crianças, jovens e adultos) com deficiência intelectual ou surdez.

Nos testes informais realizados durante apresentações em escolas e instituições não foram recolhidas métricas dos utilizadores, o público-alvo eram crianças entre 4 e 8 anos, geralmente com dificuldade na escrita, fala, audição, problemas de interação social ou que não sabem ler, ou escrever.

A maneira utilizada para medir o grau de satisfação sobre a usabilidade foram através do número de instalações, comentários dos utilizadores na Play Store e a opinião obtida por profissionais, análises realizadas em websites especializados na reabilitação e educação especial.

5.2 Publicações e Resultados

A aplicação foi publicada em três lojas de distribuição para a plataforma Android, Google Play Store, SlideME e Aptoide. A seguir está uma breve explicação sobre estas lojas e a quantidade de instalações obtidas pela aplicação desde o lançamento até o mês de setembro de 2018:

- **Google Play Store**³: É a loja oficial para distribuição e venda de produtos digitais (aplicações, jogos, música, filme e livro) da plataforma Android, está disponível para todos os dispositivos certificados, mas não está disponível em todos os países devido às políticas de censura e de embargos.

Total de instalações: 5.873

- **SlideME**⁴: É uma loja alternativa para a plataforma Android, oferece uma maior flexibilidade para o desenvolvedor e utilizador. Dispõe meios para ultrapassar os bloqueios geográficos, não possui a taxa de pagamento para o cadastro e para as publicações, oferece métodos de pagamento simplificados como o PayPal, além de possibilitar a instalação em dispositivos não certificados pelo Android.

Total de instalações: 1.930

² **Avaliação Heurística**: <https://www.nngroup.com/articles/ten-usability-heuristics/>, acessado em 29/04/2019

³ **Play Store**: <https://play.google.com/store/apps/details?id=toymobi.book1.main>, acessado em 20/12/2018

⁴ **SlideME**: <http://slideme.org/application/o-menino-que-tinha-medo-do-escuro>, acessado em 20/12/2018

- **Aptoide**⁵: É uma loja alternativa de aplicações para a plataforma Android, é uma empresa portuguesa. Está direcionada para mercados emergentes como o Brasil, Índia, México e Venezuela, além da Europa e América do Norte, oferece métodos de pagamento alternativos e não tem restrições geográficas.

Total de instalações: 1.674

As estatísticas sobre o total de instalações foram satisfatórias, mesmo sem recursos para investir na divulgação foi possível alcançar um bom número de utilizadores.

As informações foram obtidas para verificar as avaliações dos utilizadores, além de uma análise técnica sobre os modelos de dispositivos, fabricantes, versões do sistema operativo Android e quais países onde a aplicação foi instalada. As tabelas 5.2, 5.3 e 5.4 demonstram as principais métricas (total de instalações por país, modelo de dispositivo e versão do sistema operativo Android) da aplicação na plataforma de distribuição Google Play Store obtidas até o mês de Setembro de 2018.

Tabela 5.2: Instalações realizadas por país

País	Total
Brasil	4.087
Portugal	1.546
Reino Unido	51
Estados Unidos	37
África do Sul	27
Bélgica	26
Canadá	26
Cabo Verde	26
Suécia	26
Moçambique	19
Rússia	2
Total	5.873

Tabela 5.3: Instalações por versão do Android

Sistema Operativo	Total
Android 4.4	1.231
Android 7.0	1.061
Android 6.0	810
Android 5.1	585
Android 7.1	444
Android 4.2	392
Android 4.1	322
Android 5.0	306
Android 8.0	254
Android 4.0.3 até Android 4.0.4	239
Android 8.1	79
Android 2.3.3 até Android 2.3.7	75
Android 3.2	31
Android 2.2	26
Android 3.1	18
Total	5.873

Tabela 5.4: Instalações por modelo do dispositivo

Modelo	Total
Galaxy Tab E	207
Galaxy Tab 3	201
Galaxy J7 Prime	150
Galaxy Note 10.1	131
Motorola Moto G5S	124
Galaxy Tab A 7.0	108
Galaxy Tab 3 7.0 Lite	105
Galaxy J5	104
Galaxy J5 Lite	100
Galaxy J5 Prime	85
Galaxy J2 Prime TV	81
SmartLike LG MLV5	80
Galaxy S8	79
Galaxy Tab 3 7.0	77
LG G Pad V480 8.0	76
Galaxy Tab A 9.7	75
Galaxy Tab 2 7.0	71
Galaxy Tab 2 10.1	68
Asus ZenPad 10	67
Galaxy Tab 2 10.1 (P5110)	66
Galaxy Tab (P7500)	66
Motorola Moto G5S Plus	58
Outros	3.694
Total	5.873

⁵ **Aptoide**: <https://o-menino-que-tinha-medo-do.aptoide.pt/>, acessado em 20/12/2018

Foram realizadas 74 avaliações pelos utilizadores na Google Play Store, com a média de 4,770 pontos (numa escala de 1 a 5), sendo que 29 das avaliações contém o comentário do utilizador. Os principais comentários estão na tabela 5.5:

Tabela 5.5: Principais comentários das avaliações realizadas pelos utilizadores

Exemplos de comentário dos utilizadores
“Muito bom ToyMobi muito bom recomendo”
“Demora para baixar mais vale apenas abaixar já baixei uma vez e to baixando de novo!!!!”
“ Amei Parabéns pelo jogo ToyMobi”
“Parabéns pela iniciativa”
“Amo esse jogo”
“Adorei, vou usar com meus alunos na sala de recursos e recomendar as minhas colegas. Tomara que venham mais histórias.”
“Adorei! O livro é muito bom. Dá para ler com pictogramas, com gestos e normal. Parabéns! Obrigada”
“Ótimo meu irmãozinho adorou ele tem 4 aninhos”

A aplicação obteve certificação de conteúdo por diversas entidades, estando na categoria de referência, noticiosa ou educativa. Este mecanismo é muito importante para orientar os utilizadores sobre conteúdos, principalmente por se tratar de um produto destinado para o público infantil. A aplicação recebeu classificação livre para todas as idades (acima dos 3 anos) nas seguintes autoridades:

- Australian Classification Board ⁶
- Sistema de Classificação Indicativa Brasileiro ⁷
- Entertainment Software Rating Board – ESRB (América do Norte) ⁸
- Pan-European Game Information – PEGI (Europa) ⁹
- Unterhaltungssoftware Selbstkontrolle – USK (Alemanha) ¹⁰
- International Age Rating Coalition – IARC (Mundial) ¹¹

Concluiu-se que os resultados obtidos pela aplicação desenvolvida neste projeto “O menino que tinha medo do escuro” foram satisfatórios, obtendo um bom número de utilizadores, a previsão era obter 1.000 instalações e ocorreram mais de 9.000 com avaliações positivas nas lojas de aplicações.

⁶ **ACB**: <http://www.classification.gov.au>, acedido em 20/12/2018

⁷ **Classind**: <http://culturadigital.br/classind/>, acedido em 20/12/2018

⁸ **ESRB**: <http://www.esrb.org/>, acedido em 20/12/2018

⁹ **PEGI**: <http://www.pegi.info>, acedido em 20/12/2018

¹⁰ **USK**: <http://www.usk.de/en/>, acedido em 20/12/2018

¹¹ **IARC**: <https://www.globalratings.com/>, acedido em 20/12/2018

5.3 Empreendedorismo

Antes da conclusão deste projeto notou-se que era possível pensar num modelo de negócio viável, estando inserido no segmento social e com o objetivo de ser sustentável e até mesmo rentável.

O empreendedorismo requer conhecimento em diversas áreas, como gestão, marketing, contabilidade, design de produto, questões legais, entre outras. Para adquirir tais competências procurou-se apoio em programas relacionados com o empreendedorismo.

A maior parte dos programas e concursos oferecem benefícios como formações, tutoria e assistência técnica, além de outros prémios como uma bolsa monetária para desenvolver a ideia.

Este projeto participou em diversos programas e concursos de ideias de negócio, a seguir estão os principais:

16.º Concurso de Ideias – ANJE ¹²

O projeto foi selecionado como uma das 15 melhores ideias de negócio.

A entidade promotora foi a ANJE – Associação Nacional de Jovens Empresários em 2012.

O prémio foi uma ação de formação intensiva em diversas áreas relacionadas com o empreendedorismo.

Passaporte para o Empreendedorismo ¹³

O projeto foi selecionado na 1.ª fase deste programa, com a candidatura n.º 16.

O prémio era composto por uma bolsa monetária, assistência técnica e acesso a uma rede de mentores.

Este programa mudou de nome, agora chama-se Startup Voucher.

¹² **16.º Concurso de Ideias – ANJE:** <http://concursodeideias.anje.pt/site/concurso-de-ideias>, acedido em 20/10/2016

¹³ **Passaporte para o Empreendedorismo:** <http://startupportugal.com/>, acedido em 20/12/2018

Play Your Idea ¹⁴

Este concurso aconteceu em 2014 e foi promovida pela Incubadora Criativa e Tecnológica da Universidade Lusófona.

O projeto foi selecionado para a fase final do concurso.

O prémio deste concurso foi destinado apenas para os três primeiros colocados.

PAES – Programa de Apoio a Empresas Sociais ¹⁵

O projeto foi um dos 13 projetos vencedores deste programa.

Foi possível obter formação em várias áreas, tutoria e acompanhamento para elaboração do plano de negócio.

Este programa aconteceu em 2014, foi promovido pelo Banco de Inovação Social da Santa Casa da Misericórdia de Lisboa.

9.ª Edição do Concurso de Ideias de Negócio do Concelho de Cascais ¹⁶

O projeto foi premiado como a 2.ª melhor ideia na categoria empreendedorismo social.

O prémio deste concurso foi destinado apenas para o projeto vencedor entre todas as categorias.

A entidade promotora foi a associação denominada Agência DNA Cascais, esta edição do concurso ocorreu em 2015.

Arrisca C ¹⁷

O projeto foi um dos 10 finalistas na categoria ideia de negócios.

Foi um grande reconhecimento estar entre os finalistas, trata-se de um dos concursos mais concorridos e respeitados em Portugal.

Este concurso ocorreu em 2016, foi promovido por um conjunto de entidades liderados pela Universidade de Coimbra e com o apoio de 22 patrocinadores.

¹⁴ **Play Your Idea:** <http://play.ulusofona.pt/index.php/eventos/44-concurso-play-your-idea>, acedido em 20/11/2016

¹⁵ **PAES – Programa de Apoio a Empresas Sociais:** <https://bancodeinovacaosocial.eu/portfolio-item/paes-2/>, acedido em 20/12/2017

¹⁶ **9.ª CINC.:** <http://www.dnacascais.pt/ecossistema-empendedor/ideias-negocios-e-internacionalizacao/dna-ideias-e-negocios/>, acedido em 20/12/2017

¹⁷ **Arrisca C:** http://www.uc.pt/gats/eventos_e_iniciativas/a_decorrer/arrisca_c, acedido em 20/12/2018

Empreende+¹⁸

O projeto foi selecionado como um dos 10 finalistas neste programa.

Esta iniciativa ofereceu uma bolsa monetária, sessões de formações e tutoria em diversas áreas.

Foi promovido pela associação Jovens Empresários e Empreendedores Católicos – JEEC–AN, durante o ano de 2016.

Wanted Business Ideas¹⁹

O projeto foi um dos 12 finalistas neste concurso.

Este concurso disponibilizou diversas oficinas de criatividade e geração de ideias, um programa de capacitação, consultoria e tutoria.

Foi promovido pela Rede Regional de Empreendedorismo Viseu Dão Lafões em 2017.

1.º Concurso de Ideias Empreender Leiria²⁰

O projeto foi eleito o vencedor da primeira edição deste concurso.

O concurso facultou um prémio monetário para o desenvolvimento da ideia.

Foi organizado pela Associação Empresarial da Região de Leiria – NERLEI em 2017.

Concurso Bolsas+²¹

O projeto foi um dos selecionados como finalista.

O prémio foi uma quantia monetária para o desenvolvimento da ideia, além de um ano de alojamento grátis numa das incubadoras da região.

Foi organizado pela Rede de Empreendedorismo das Beiras e Serra da Estrela – REBSE.

Outro grande reconhecimento para este projeto foi ingressar na incubadora de empresas do Instituto Pedro Nunes, criada pela Universidade de Coimbra para promover a inovação, investigação e o desenvolvimento tecnológico. Foi eleita em 2010 a melhor incubadora de base tecnológica pelo concurso mundial “Best Science Based Incubator”²².

¹⁸ **Empreende+:** <http://www.empreendemaais.pt/>, acedido em 20/12/2018

¹⁹ **Wanted Business Ideas:** <http://empreendeviseudaolafoes.pt/wanted/>, acedido em 20/12/2018

²⁰ **1.º Concurso de Ideias Empreender Leiria:** <http://empreenderleiria.pt/pt/concurso>, acedido em 11/11/2018

²¹ **Concurso Bolsas +:** <http://www.empreenderecrescer.pt/concurso-bolsas>, acedido em 20/12/2018

²² **Best Science Based Incubator:** <https://www.ipn.pt/noticias/noticia/1857>, acedido em 17/02/2016

5.4 Divulgações

Durante o desenvolvimento deste projeto foram publicados resumos e artigos em eventos relacionados com a educação e tecnologia. O primeiro foi no “2.º Simpósio do Mestrado em Computação Móvel”²³ e o segundo foi na “II Jornadas de Educação: Desafios no 1.º CEB”²⁴ ambos organizados pelo Instituto Politécnico da Guarda. O último foi durante o evento internacional “4.º Encontro sobre Jogos e Mobile Learning”²⁵, realizado pela Universidade de Coimbra e pela Universidade de Sevilha.

Este projeto foi apresentado em diversas instituições de ensino e instituições que cuidam de pessoas com deficiência com o objetivo em demonstrar e testar a aplicação.

Houve uma grande aceitação do projeto e da aplicação desenvolvida, principalmente nos testes com crianças e pessoas com deficiência, além de profissionais como educadores, técnicos em educação especial e terapeutas.

Durante o processo das apresentações em instituições, programas de empreendedorismo e nos concursos de ideias deu origem a diversas divulgações nos meios de comunicação, conforme demonstrado a seguir:

Reportagem sobre o livro inclusivo “O menino que tinha medo do escuro”

Esta reportagem foi realizada no dia 03 de dezembro de 2014, pelo programa Portugal em Direto da emissora RTP. Ocorreu durante a comemoração do Dia Internacional das Pessoas com Deficiência. O local da reportagem foi no Jardim de Infância de Maçainhas, freguesia do concelho da Guarda, como demonstra a figura 5.3.

²³ **2.º Simpósio do Mestrado em Computação Móvel:** <http://mcm.ipg.pt/doku.php?id=programa2012>, acedido em 20/12/2018

²⁴ **II Jornadas de Educação: Desafios no 1.º CEB”:** <http://www.ipg.pt/2jornadaseducacao/programa.aspx>, acedido em 20/12/2018

²⁵ **4.º Encontro sobre Jogos e Mobile Learning:** <https://estudogeral.sib.uc.pt/handle/10316/48542>, acedido em 20/12/2018



Figura 5.3: Reportagem no Programa Portugal em Direto – RTP ²⁶

Apresentação para alunos com surdez

Foi realizada no dia 12 de janeiro de 2015 em Lisboa, no Agrupamento de Escolas Quinta de Marrocos, Benfica. A apresentação foi para uma turma de alunos do 3.º, 4.º e 5.º anos com surdez ou deficiência auditiva, como demonstra a figura 5.4.



Figura 5.4: Apresentação no Agrupamento de Escolas Quinta de Marrocos²⁷

²⁶ Reportagem no Programa Portugal em Direto:: <https://youtu.be/wXKUtgfHkuw>, acedido em 20/12/2018

²⁷ Vídeo nas Escolas Quinta de Marrocos: <https://youtu.be/y-ip3ZbEHlg>, acedido em 20/12/2018

Reportagem do Jornal O Interior – Guarda

Esta notícia foi publicada no dia 08 de janeiro de 2015, no formato digital e no jornal impresso, como demonstra a figura 5.5.

Guarda

Lançada aplicação para pessoas com necessidades especiais

Mateus Victorelli, aluno do Mestrado em Computação Móvel no Instituto Politécnico da Guarda, acaba de lançar uma aplicação para pessoas com necessidades especiais, baseada no livro infantil inclusivo "O menino que tinha medo do escuro", da autoria de Susana Campos.



para dispositivos Android "O menino que tinha medo do escuro" é totalmente gratuita e sem nenhuma publicidade. Oferece três modos para leitura e interpretação da história: Texto, Ilustração e Narração; Texto, Sistema SPC (Símbolos Pictográficos para Comunicação) e conversão dos símbolos pictográficos para voz; Língua Gestual Portuguesa. Oferece três jogos: Memória, com seis níveis de dificuldade; Pintura livre ou com formas geométricas; Quebra-Cabeça/Puzzle. O link na Loja de Aplicações Android (Google Play Store) é <https://play.google.com/store/apps/details?id=toymobi.book1.main>.

Figura 5.5: Reportagem no jornal O Interior

Publicada no dia 12 de março de 2015 no jornal impresso, como demonstra a figura 5.6.

Projecto ToyMobi premiado pelo PAES Invest

Mateus Victorrelli *“não esperava que fosse dar tão certo”* a candidatura ao Programa de Apoio às Empresas Sociais (PAES), do Banco de Inovação Social da Santa Casa da Misericórdia de Lisboa. Denominado PAES Invest, o programa, que conta com a parceria da Santa Casa da Misericórdia do Porto, é dirigido aos empreendedores que oferecem uma resposta inovadora, sustentável e geradora de valor social e económico, para problemas ou necessidades sociais.

Dos 349 projectos candidatados, foram escolhidos 13 finalistas, aos quais será disponibilizada uma linha de crédito até aos 10 mil euros.

ToyMobi foi um dos premiados. O promotor é Mateus Victorrelli, brasileiro a residir na Guarda há quatro anos, aluno do Mestrado em Computação Móvel no Instituto Politécnico da cidade, que tem como parceiras a guardense Susana Campos, autora de “O menino que tinha medo do escuro”, que serve como estímulo para atitudes mais inclusivas e criar

conteúdos acessíveis a crianças com necessidades específicas, e Fernando Bastos, natural do Fundão, professora de Educação Especial desde 1990.

A ideia do projecto partiu de Mateus Victorrelli - “é a minha tese de mestrado” -, tendo como finalidade “o desenvolvimento de uma solução tecnológica para suprir uma necessidade social, mais especificamente desenvolver jogos para pessoas com défice cognitivo, utilizando como plataforma os dispositivos móveis”.

Mas, “após uma análise sobre os produtos existentes no mercado, ficou comprovado a existência de uma grande carência de produtos para um público-alvo específico (pessoas com necessidades educativas especiais e pessoas com algum tipo de défice), motivados por diversos problemas sociais e culturais, e pelo mercado que tem uma enorme dificuldade em compreender as necessidades destas pessoas”, afirma.

Motivo que levou a que a ideia fosse alterada “para abranger não apenas as pessoas com défice cognitivo” como também pessoas “com necessidades especiais”.

Mudança que se deve também, destaca, ao facto de ter conhecido Susana Campos.

“Ideias não faltam”

“Temos três gamas de produtos pré-definida. Uma já está lançada, que é o livro multimédia inclusivo, baseado na história “O menino que tinha medo do escuro”, que tinha como objectivo, segundo a Susana Campos, concretizar a ideia de Mateus, adaptando que a autora tem “outro livro que está em fase de ilustração”, “outra serão os jogos lúdico-pedagógicos, que vamos converter para a parte digital, e, mais volada para a psicologia, uma plataforma de execução e acompanhamento de testes psicológicos e pedagógicos”. “Fiz algumas entrevistas com psicólogos e educadores, e a coisa que achei mais absurda é que a maior parte dos testes realizados para crianças são em papel. E isso se para qualquer pessoa é chato, então para criança nem se fala”, sublinha.

“Ideias não faltam”, garante Mateus Victorrelli, dizendo que “queremos consolidar pelo menos estas três gamas para depois expandir ainda mais”. “Temos ideias utilizando robótica, electrónica, realidade aumentada. Temos uma gama de produtos enorme mas queremos um passo de cada vez para não correremos mais riscos que aqueles que já estamos a correr”, ressalva o mentor. “O projecto simplesmente é único. Em Portugal tenho a certeza que não existe nada igual e em termos mundiais não me lembro de ter visto outro parecido”, assegura.

Um dos objectivos da equipa do ToyMob é criar uma empresa, e, tal como “já deixei bem específico o plano de negócios”, a sua localização “vai ser no Interior”. A Guarda é mais provável escolha, não só por ser a cidade onde vivem dois dos elementos, e estar perto do terceiro membro da equipa, mas porque a ideia é “criar mais oportunidades para tentar absorver os estudantes que saem do Politécnico e até mesmo da UBI [Universidade da Beira Interior, na Covilhã].

A linha de crédito disponibilizada pelo PAES Invest, num máximo de 10 mil euros, “são essenciais para criarmos a empresa”, confessa Mateus Victorrelli, revelando um possível interesse de alguns investidores presentes na cerimónia de apresentação dos projectos finalistas ao programa da Santa Casa da Misericórdia de Lisboa.

GM

Figura 5.6: Reportagem no jornal Terras da Beira

Reportagem do Diário de Viseu

Esta notícia foi publicada no dia 29 de junho de 2017, no jornal impresso, como demonstra a figura 5.7.

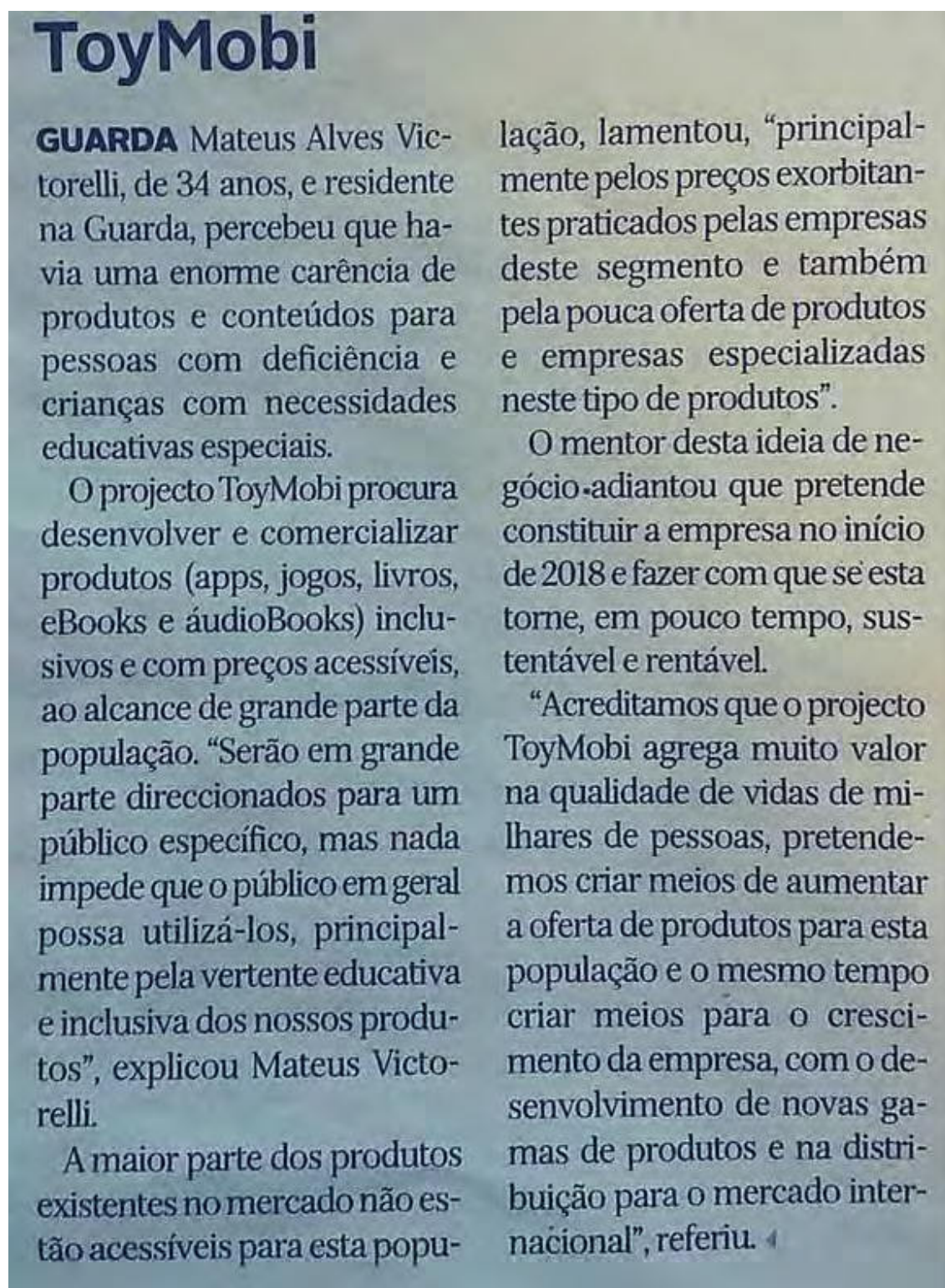


Figura 5.7: Reportagem no jornal Diário de Viseu

Centro de Recursos para a Inclusão Digital – CRID²⁸: Divulgação realizada no dia 16 outubro de 2014, pelo website do CRID.

Santa Casa da Misericórdia de Lisboa²⁹: Divulgação realizada no dia 22 outubro de 2014, pelo website e lista de e-mails da Santa Casa da Misericórdia de Lisboa.

Blog do Magazine Consigo RTP2³⁰: Publicado no dia 22 outubro de 2014, o Consigo é um programa televisivo que é resultante da parceria entre a RTP e o Instituto Nacional para a Reabilitação.

Instituto Nacional para a Reabilitação, I. P.³¹: Publicação do mês de outubro de 2014 pela Biblioteca do Instituto Nacional para a Reabilitação, I. P.

Agrupamento de Escolas Afonso de Albuquerque – Guarda³²: Publicada no dia 28 de novembro de 2014.

Blog Compartilha – Universidade de Aveiro³³: Notícia publicada no dia 18 de fevereiro de 2015.

Divulgação no Instituto Pedro Nunes³⁴: Notícia publicada no dia 14 de julho de 2015.

Portal Reab³⁵: Publicada em 30 de novembro de 2015, o Reab é considerado o maior portal sobre reabilitação no idioma português.

II Jornadas de Educação: “Desafios no 1.º CEB”³⁶: Apresentação “ToyMobi – Projetos digitais em Educação Especial”, realizada no dia 27 de outubro de 2016.

²⁸ **CRID**: <http://crid.esecs.ipleiria.pt/2014/10/16/convite-para-apresentacao-do-livro-o-menino-que-tinha-medo-do-escuro/>, acedido em 20/12/2018

²⁹ **SCML**: http://www.scml.pt/pt-PT/media/noticias/_o_menino_que_tinha_medo_do_escuro__2/, acedido em 20/12/2018

³⁰ **Consigo**: <http://magazineconsigo.blogspot.com/2014/10/livro-infantil-inclusivo-o-menino-que.html>, acedido em 20/12/2018

³¹ **Biblioteca do INR**: <http://www.inr.pt/content/1/3718/outubro-de>, acedido em 20/12/2018

³² **AEAAG**: <http://educacaoespecialaeaag.blogspot.com/2014/11/apresentacao-do-livro-o-menino-que.html>, acedido em 20/12/2018

³³ **Blog Compartilha** <http://cms.ua.pt/compartilha/node/302>, acedido em 20/12/2018

³⁴ **Instituto Pedro Nunes** <https://www.ipn.pt/noticias/noticia/1866>, acedido em 20/12/2018

³⁵ **Reab**: <https://www.reab.me/o-menino-que-tinha-medo-do-escuro-livro-inclusivo/>, acedido em 20/12/2018

³⁶ **II Jornadas de Educação**: <http://www.ipg.pt/2jornadaseducacao/>, acedido em 20/12/2018

5.5 Outros Projetos

Este projeto continua a desenvolver novas soluções para este segmento, principalmente direcionado para as crianças com necessidades especiais de educação. Foi publicada uma aplicação e outras três estão em fase de desenvolvimento, além de novas ideias para serem investigadas no futuro.

Ernesto Jogos para a Gaguez:

- É uma aplicação única no mercado, composta por um livro multimédia e jogos especialmente concebidos para crianças e jovens com gaguez.
- Foi desenvolvida em parceria com a autora e terapeuta da fala Mónica Gaiolas e a editora Coisas de Ler.
- Está disponível para a plataforma Android, distribuído com o preço de 3,49€, com suporte em inglês e português, foi lançada em Julho de 2015.
- Foram realizadas até o momento um total de 215 vendas em 16 países.
- A figura 5.8 demonstra esta aplicação.



Figura 5.8: Aplicação – Ernesto, Jogos para a Gaguez ³⁷

³⁷ **Ernesto, Jogos para a Gaguez:** <https://play.google.com/store/apps/details?id=toymobi.ernesto>, acedido em 20/12/2018

Os Guerreiros do pequeno Artur:

- Está em fase de desenvolvimento, será um conjunto de produtos (livro, e-book, áudio-livro e aplicação móvel).
- Tem como base uma história infantil que aborda uma doença crónica, a esclerose múltipla.
- Trata-se de uma história infantil única, não existe nada similar no mercado, aborda a vida de uma criança portadora desta doença.
- Estará disponível em português, inglês, espanhol, francês, alemão, italiano, polaco, turco, russo, hindi, árabe, mandarim e japonês.
- A figura 5.9 demonstra esta aplicação.



Figura 5.9: Aplicação – Os Guerreiros do Pequeno Artur

Prancha Inclusiva: Estações do Ano:

- Será uma aplicação móvel com elementos da comunicação alternativa e aumentativa, com o complemento da tradução para a língua gestual portuguesa.
- Contém um conjunto de itens relacionados com as estações do ano e elementos característicos de Portugal, como os frutos, alimentos, datas comemorativas e os jogos tradicionais.
- A figura 5.10 demonstra esta aplicação.

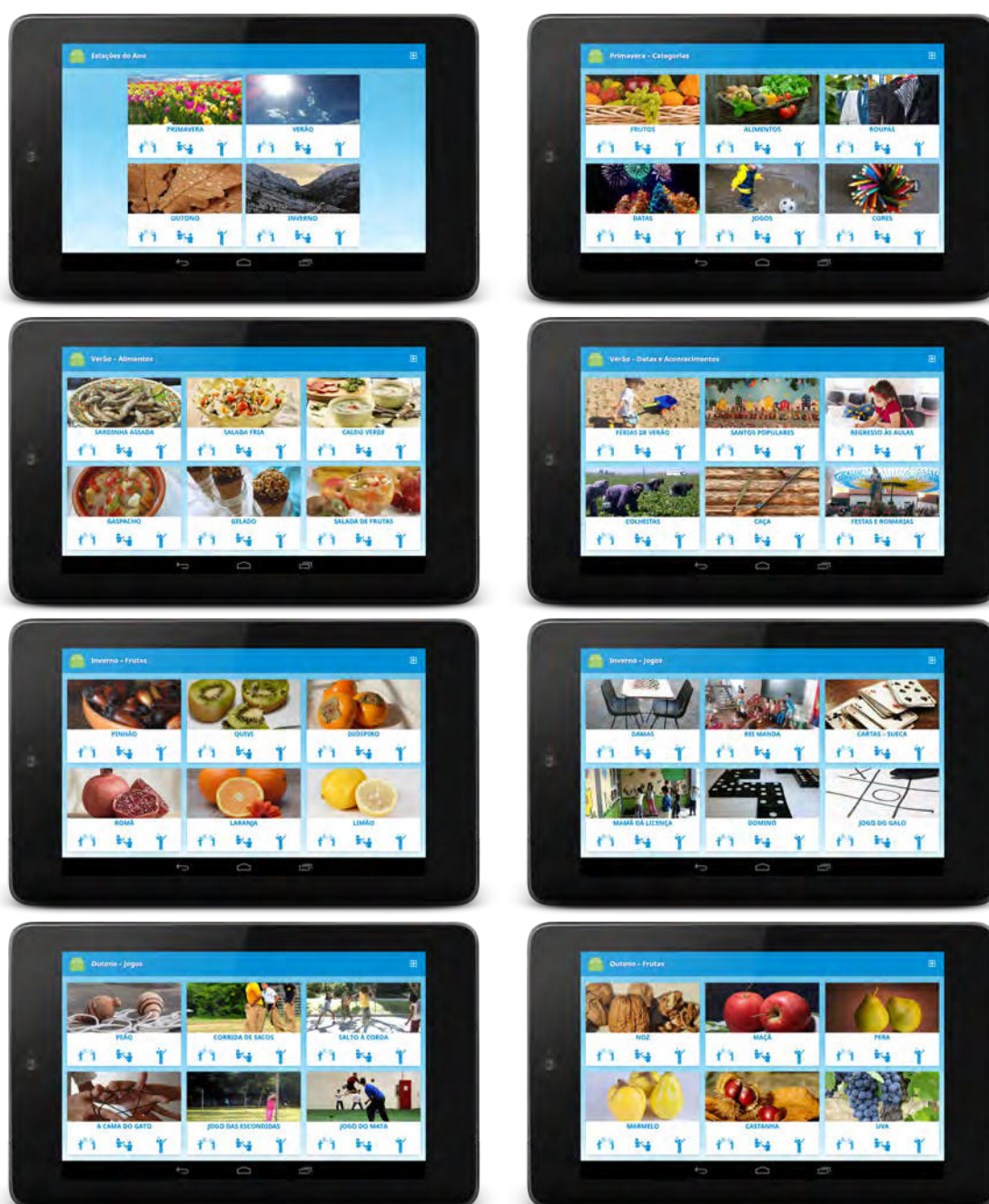


Figura 5.10: Aplicação – Prancha Inclusiva Estações do Ano

Capítulo 6

Conclusão e Trabalho Futuro

Neste capítulo são apresentados os resultados que foram obtidos e mostra os planos para o trabalho futuro em relação à aplicação “O menino que tinha medo do escuro”. Além de apresentar outras ideias para investigação e desenvolvimento de novas soluções nesta área.

6.1 Conclusão

Acreditamos que este projeto alcançou os objetivos e expectativas propostos, demonstrou-se o benefício das aplicações para os dispositivos móveis como ferramentas acessíveis e adequadas para a educação e inclusão, principalmente direcionadas para as crianças na educação especial e até mesmo adultos com alguma deficiência ou limitação. Este modelo de aplicações pode ajudar no trabalho de profissionais da educação, terapias, psicologia e até mesmo os familiares das pessoas com limitações.

Este projeto ultrapassar diversas barreiras, sobretudo relacionadas com a pouca oferta de produtos digitais inclusivos destinados para as pessoas com deficiência ou limitação e principalmente para as crianças na educação especial.

A solução desenvolvida procurou alcançar objetivos sociais com a computação móvel. Englobando desde os temas relacionados com a deficiência, educação especial e a inclusão em conjunto com os conceitos de desenvolvimento para as plataformas móveis e o empreendedorismo.

Os resultados obtidos pela aplicação desenvolvida neste projeto “O menino que tinha medo do escuro” foram positivos. Foi possível alcançar um bom número de utilizadores e mesmo estando disponível apenas em português ocorreram instalações em diversos países com outra língua oficial. Foram realizadas 74 avaliações pelos utilizadores na loja de aplicações Google Play Store, com a média de 4,742 pontos (numa escala de 1 a 5), sendo que 23 das avaliações contêm o comentário positivo.

Na vertente do empreendedorismo, este projeto resultou na criação de uma empresa com o objeto social inserido na investigação e desenvolvimento de produtos destinados para as Tecnologias de Informação e Comunicação em Educação (TICEDU), a venda e distribuição de produtos digitais (aplicações, jogos e livros digitais) e de produtos físicos (livros, jogos e materiais didáticos impressos), além de formação e assistência técnica.

Todo o conteúdo elaborado neste projeto está disponível num repositório online¹, usado no controlo de versão do código-fonte e alojamento dos materiais produzidos neste projeto aplicado.

6.2 Trabalho Futuro

Em relação ao trabalho futuro, pretende-se realizar atualizações na aplicação “O menino que tinha medo do escuro”, introduzindo novas funcionalidades e processos, tais como:

- Realizar testes de usabilidade destinados para a educação especial.
- Traduzir para a língua brasileira de sinais (LIBRAS).
- Traduzir a história para inglês e espanhol.
- Adaptar para os símbolos pictográficos de comunicação para inglês e espanhol.
- Traduzir para língua gestual espanhola e americana.
- Inserir a opção de gravar na persistência uma determinada imagem feita no jogo de desenhar.
- Alterar o formato do jogo quebra-cabeça, adicionando uma opção para diminuir a quantidade de peças.
- Aumentar o nível de dificuldade no jogo de completar palavras.
- Adicionar novos jogos lúdico-pedagógicos.
- Atualizar o código-fonte para utilizar a versão mais recente do Android.

¹ **Repositório:** https://bitbucket.org/account/user/mcm_ipg_2017/projects/MES, acedido em 07/05/2019

Existem outras ideias que serão investigadas, visto que este projeto pretende continuar na pesquisa e desenvolvimento de novas soluções tecnológicas nesta área. A seguir são apresentados exemplos de temas que estão em fase de investigação;

- **Impressão 3d:** Confeccionar um conjunto de brinquedos educacionais adaptados para pessoas com deficiência, principalmente com deficiência visual.
- **Óculos de Realidade Virtual e Aumentada:** Criar aplicações móveis que utilizem a realidade virtual e aumentada no contexto educacional, na reabilitação cognitiva e motora.
- **Smartwatch:** Desenvolver aplicações para os relógios inteligentes, utilizadas para auxiliar a reabilitação motora.
- **Internet das Coisas (IoT):** Idealizar um novo ambiente educacional onde o espaço físico (sala de aula) interage com os utilizadores (alunos e professores) através dos equipamentos tecnológicos (IoT).

Bibliografia

- [Ahmad et al., 2018] Ahmad, N., Rextin, A., and Kulsoom, U. E. (2018). Perspectives on Usability Guidelines for Smartphone Applications: An Empirical Investigation and Systematic Literature Review. *Information & Software Technology*, 94:130 – 149. 16
- [Almeida e Tavares, 2010] Almeida, M. M. e Tavares, H. M. (2010). Síndrome de williams e a intervenção da psicomotricidade com auxílio na escolarização. *Uberlândia, Revista da Católica*, 2 (3), 2(03):334–347. 53
- [Alqahtani e Mohammad, 2015] Alqahtani, M. e Mohammad, H. (2015). Mobile Applications’ Impact on Student Performance and Satisfaction. *Turkish Online Journal of Educational Technology - TOJET*, 14(4):102 – 112. 13, 19
- [Anh, 2016] Anh, H. N. (2016). Smartphone Industry: the New Era of Competition and Strategy. *Centria University of Applied Sciences*, página 43. 16
- [Association, 2016] Association, G. (2016). The Mobile Economy 2016. 17
- [Assunção, 2014] Assunção, J. C. (2014). Surdez em Portugal: Estudos Genéticos e Funcionais do Gene da Conexina. 53, 54
- [Azevedo, 2005] Azevedo, L. (2005). As Tecnologias de Informação e Comunicação e as Necessidades Especiais. *Revista Diversidades*, (7):4–9. 9, 10
- [Azevedo, 2011] Azevedo, M. L. B. (2011). A Inclusão de uma Criança com Síndrome do X Frágil – Estudo de Caso. 52
- [Behdioglu e Sener, 2014] Behdioglu, S. e Sener, H. Y. (2014). Improving Service Quality in Special Education Institutions: Servqual Scale. *Global Business And Management Research*, 6(2):169 – 184.
- [Campina, 2016] Campina, A. L. P. (2016). Campina, A. L. P. (2016). Sonhar, Planear e Concretizar Introdução da Comunicação Aumentativa e Alternativa através dos Símbolos Pictográficos de Comunicação. 9, 10
- [Carey, 2015] Carey, R. A. (2015). The Game Design Canvas - Serious Play Conference. 62
- [DGEEC, 2016] DGEEC (2016). Questionário das necessidades especiais de educação: Ano letivo 2015/2016. 50

- [Diário da República, 2004] Diário da República (2004). Bases Gerais do Regime Jurídico da Prevenção, Habilitação, Reabilitação e Participação da Pessoa com Deficiência. *Diário da República - I Série-A Nº 194 - 18 Agosto de 2004*, 1(194):5232–5236. 48
- [Dutta et al., 2017] Dutta, A., Puvvala, A., Roy, R., and Seetharaman, P. (2017). Technology diffusion: Shift happens – The case of iOS and Android handsets. *Technological Forecasting Social Change*, 118:28 – 43. 19
- [Feijão, 2015] Feijão, T. M. N. (2015). A Inclusão do Portador de Deficiência Auditiva na Educação Regular: Métodos de Comunicação a Utilizar. 53
- [Ferreira, 2017] Ferreira, M. G. (2017). Um Olhar Mediador sobre a Deficiência e a Formação Profissional – O Caso do Centro de Formação Profissional do Centro de Educação do Cidadão Deficiente de Mira Sintra. 3
- [Figueiredo, 2015] Figueiredo, C. (2015). Mercado de Produtos Tecnológicos em Portugal. 18
- [França, 2008] França, A., N. C. M. D. . A. (2008). Abordagem Pedagógico Educativa: Um Percurso. *Revista Diversidades*, 22(4–9). 51
- [GEP, 2016] Gabinete de Estratégia e Planeamento (2016). Estatísticas sobre Deficiências ou Incapacidades. 3, 7, 49, 50
- [Istenic Starcic e Bagon, 2014] Istenic Starcic, A. e Bagon, S. (2014). Ict-supported Learning for Inclusion of People with Special Needs: Review of Seven Educational Technology Journals, 1970-2011. *British Journal of Educational Technology*, 45(2):202 – 230. 9
- [Kuo et al., 2007] Kuo, J. Y., Ditchekenian, V., Manna, T. D., Kuperman, H., Damiani, D., and Setian, N. (2007). Síndrome de Prader-Willi: Aspectos Metabólicos Associados ao Tratamento com Hormônio de Crescimento. *Arquivos Brasileiros de Endocrinologia & Metabologia*, 51(1):92–98. 53
- [Lewis, 2014] Lewis, J. (2014). Microservices – A Definition of this New Architectural Term. 64, 65, 66
- [Maris e Trott, 2011] Maris, A. F. e Trott, A. (2011). A Patogênese Genética e Molecular da Síndrome de Angelman. *Jornal Brasileiro de Psiquiatria*, 60(4):321–330. 53
- [Oliveira, 2013] Oliveira, M. F. S. B. d. (2013). Construção de um Programa de Enriquecimento e Estimulação do Potencial Cognitivo – PEEC: Aplicação à População Escolar com Défice Cognitivo Moderado. 7, 56, 57

- [Organização Mundial da Saúde, 2012] Organização Mundial de Saúde (2012). Relatório Mundial sobre a Deficiência. *Journal of Applied Research in Intellectual Disabilities*, 25(6). 2, 47, 48
- [Pestana, 2011] Pestana, M. n. G. (2011). Deficiência intelectual: Transição para a vida activa. 51
- [Quelhas e Mesquita, 2011] Quelhas, M. d. R. e Mesquita, M. H. (2011). O Uso das TIC por Jovens Portadores de T21: Um Estudo de Caso. *CIDTFF – Indagatio Didactica – Universidade de Aveiro*, 3 (2)(1647-3582):92–112. 52
- [Richardson e Smith, 2016] Richardson, C. e Smith, F. (2016). Microservices – From Design to Deployment. *Nginx*. 64
- [Érika Rímoli Mota da e Maria da Piedade Resende da, 2018] Érika Rímoli Mota da, S. e Maria da Piedade Resende da, C. (2018). Revisão Sistemática sobre Avaliação para Identificação Inicial de Alunos com Deficiência Intelectual. *Revista Educação Especial*, Vol 31, Iss 62, Pp 551-568 (2018), (62):551.
- [Rodrigo Sousa de et al., 2014] Rodrigo Sousa de, M., Carla Oliveira, S., and Wiliam César Alves, M. (2014). Communication with People with Hearing Disabilities: An Integrative Review. *Revista de Pesquisa : Cuidado é Fundamental Online*, Vol 6, Iss 4, Pp 1695-1706 (2014), (4):1695.
- [Rodrigues e Nogueira, 2010] Rodrigues, D. e Nogueira, J. (2010). Educação Especial e Inclusiva em Portugal Factos e Opções. *Revista de Educación Inclusiva*, 3(1):97–109. 8
- [Rogers, 2016] Rogers, W. (2016). Global Smartphone Sales Hit a Quarterly High in Q4 2015. 17
- [Saha e Saha, 2018] Saha, A. K. e Saha, A. (2018). Smartphone as a Means of Communication & its Influence Upon College Students. *Indian Journal of Industrial Relations*, 53(4):560 – 571. 13
- [Sasaki, 2005] Sasaki, R. K. (2005). Atualizações Semânticas na Inclusão de Pessoas: Deficiência Mental ou Intelectual? Doença ou Transtorno Mental? *Revista Nacional de Reabilitação*, ano IX(43):9–10. 51
- [Shen et al., 2017] Shen, S., Lu, X., and Hu, Z. (2017). Towards Release Strategy Optimization for Apps in Google Play. 16
- [Silva, 2017] Silva, A. (2017). Projecto-lei n.º 329/xiii/2 – Acessibilidade Efectiva para Todos os Estudantes com Necessidades Educativas Especiais no Ensino Superior. página 7. 3

- [Sousa e dos Santos, 2005] Sousa, P. M. L. e dos Santos, I. M. S. C. (2005). Caracterização da Síndrome Autista. *O Portal dos Psicólogos*. 52
- [Trabelsi et al., 2017] Trabelsi, Z., Al Matrooshi, M., Al Baira, S., Ibrahim, W., and Masud, M. M. (2017). Android Based Mobile Apps for Information Security Hands-on Education. *Education and Information Technologies*, 22(1):125 – 144. 17, 19
- [Tripoli e Carvalho, 2016] Tripoli, C. d. S. e Carvalho, R. P. (2016). Microserviços: Características, Benefícios e Desvantagens em Relação à Arquitetura Monolítica que Impactam na Decisão do Uso desta Arquitetura. 65
- [UNESCO, 1994] UNESCO (1994). Declaração de Salamanca e Enquadramento da Ação na Área das Necessidades Educativas Especiais. *Conferência Mundial Sobre Necessidades Educativas Especiais: Acesso e Qualidade*. 51
- [Watkins, 2013] Watkins, A. (2013). Tecnologias de Informação e Comunicação para a Inclusão: Desenvolvimentos e Oportunidades para Países Europeus. 9